

RUBY-D714VG2AR

Industrial Motherboard
ATX Board

User's Manual

Version 1.0

Copyright © Portwell, Inc., 2013 All rights reserved.
All other brand names are registered trademarks of their respective owners.

Table of Contents

How to Use This Manual

Chapter 1 System Overview	1-1
1.1 Introduction	1-1
1.2 Check List	1-1
1.3 Product Specification	1-2
1.3.1 Mechanical Drawing	1-4
1.4 System Architecture	1-5
Chapter 2 Hardware Configuration	2-1
2.1 Jumpers	2-1
2.2 Connector Allocation	2-6
Chapter 3 System Installation	3-1
3.1 Intel® Dual Core/Quad Core processor	3-1
3.2 Main Memory	3-5
3.3 Installing the Single Board Computer	3-6
3.3.1 Chipset Component Driver	3-6
3.3.2 Intel® HD Graphics Family	3-6
3.3.3 Intel® PROSet Gigabit Ethernet Controlle	3-6
3.3.4 Audio Controller	3-7
3.3.5 Intel® Active Management Technology (Intel® AMT)	3-7
3.4 Clear CMOS Operation	3-7
3.5 WDT Function	3-7
3.6 GPIO	3-9
Chapter 4 BIOS Setup Information	4-1
4.1 Entering Setup -- Launch System Setup	4-1
4.2 Main	4-2
4.3 Configuration	4-3
4.4 Security	4-26
4.5 Boot	4-27
4.6 Exit	4-28
Chapter 5 Troubleshooting	5-1
5.1 Hardware Quick Installation	5-1
5.2 BIOS Setting	5-3
5.3 FAQ	5-5
5.4 System Memory Address Map	5-6

Appendix A

Appendix B

How to Use This Manual

The manual describes how to configure your RUBY-D714VG2AR system board to meet various operating requirements. It is divided into five chapters, with each chapter addressing a basic concept and operation of Single Host Board.

Chapter 1: System Overview. Presents what you have in the box and give you an overview of the product specifications and basic system architecture for this series model of single host board.

Chapter 2: Hardware Configuration. Show the definitions and locations of Jumpers and Connectors that you can easily configure your system.

Chapter 3: System Installation. Describes how to properly mount the CPU, main memory and Compact Flash to get a safe installation and provides a programming guide of Watch Dog Timer function.

Chapter 4: BIOS Setup Information. Specifies the meaning of each setup parameters, how to get advanced BIOS performance and update new BIOS. In addition, POST checkpoint list will give users some guidelines of trouble-shooting.

Chapter 5: Troubleshooting. Provide various of useful tips to quickly get RUBY-D714VG2AR running with success. As basic hardware installation has been addressed in Chapter 3, this chapter will basically focus on system integration issues, in terms of backplane setup, BIOS setting, and OS diagnostics.

The content of this manual is subject to change without prior notice. These changes will be incorporated in new editions of the document. The vendor may make supplement or change in the products described in this document at any time

Chapter 1

System Overview

1.1 Introduction

Powell Inc., a world-leading innovator in the Industrial PC (IPC) market and a member of the Intel® Communications Alliance, has launched its new RUBY-D714VG2AR in response to market demand for a simplified embedded system board (ESB) that combines a smaller footprint, lower power consumption, robust computing power and with longevity support.

Against the pre-generation chipset RUBY-D714VG2AR based on Q67 chipset, Q77 offers Core i5/i7 processors, VGA, DVI-D, Display port, USB 3.0 and SATA III support which USB 3.0 is first integrated.

RUBY-D714VG2AR based on the latest 3rd Gen Intel® Core™ Processor which offers 22nm Hi-K process technology with energy efficient architecture. RUBY-D714VG2AR adopts four channels DDR3 long DIMMs up to 32GB.

Desktop solution is still popular in the market of DVR and Factory Automation which can fulfill most of these applications; therefore, with high performance and high-end specifications, Q77 is our first generation Ivy bridge chip architecture on ATX line.

1.2 Check List

The RUBY-D714VG2AR package should cover the following basic items

- ✓ One RUBY-D714VG2AR Industrial Mother board
- ✓ One SATA III Cable
- ✓ One I/O Shield bracket
- ✓ One Installation Resources CD-Title

If any of these items is damaged or missing, please contact your vendor and keep all packing materials for future replacement and maintenance.

1.3 Product Specification

- **Main Processor**

Intel® Dual Core/Quad Core LGA1155 processor

- **Chipset**

Intel® Q77 Express chipset

- **System BIOS**

Phoenix UEFI BIOS

- **Main Memory**

-Four 240-pin DDR3 DIMM socket support up to 32GB dual channel 1600/1333MHz memory

-RUBY-D714VG2AR supports Non-ECC memory

- **Expansion Interface**

One PCIe x16 slot

One PCIe x16 slot (PCIe x4 signal)

One PCIe x4 slot (PCIe x1 signal)

Four PCI slots

- **SATA Interface**

Five SATA ports(Two SATA 6Gb/s, Three SATA 3Gb/s)

One CFEX (SATA 3Gb/s interface)

- **Serial Port**

Support four RS232 and two RS232/422/485

- **USB Interface**

Support Ten USB ports, four on rear I/O and six on board header for internal devices

- **Audio Interface**

Connector for Mic-In, Line-In and Line-Out

- **Real Time Clock/Calendar (RTC)**

Support Y2K Real Time Clock/Calendar

- **Watch Dog Timer**

Support WDT function through software programming for enable/disable and interval setting

General system reset

- **On-board Ethernet LAN**

Two Gigabit Ethernet (10/100/1000 Mbits/sec) LAN ports using Intel 82579LM & 82583V GbE Ethernet Controller

- **High Drive GPIO**

One pin-header for 8 bit GPIO(4bit in & 4bit out)

- **System Monitoring Feature**

Monitor system temperature and major power sources.

- **Outline Dimension (L x W)**

304.8mm (12") X 243.8mm (9.6")

- **Power Requirements**

Item	Power ON	Full Loading 10Min	Full Loading 30Min
CPU +12V	1.78	4.67	4.10
System +12V	0.37	0.7	0.73
System +3.3V	0.50	0.81	0.83
System +5V	1.45	2.04	1.88
System+ Device +12V	0.66	0.75	0.78
System+ Device +5V	1.46	2.06	1.95
USB Loading Test	4.65-4.76V/ 460-510 mA		

- **Configuration:**

CPU Type	Intel® Core™ i7-3770 CPU@3.40G Hz L3:8M
SBC BIOS	Portwell, Inc.RUBY-D714 VG2AR TEST BIOS(20705T00)
Memory	Transcend DDR3 1333 1G*2 (SEC HCH9K4B1G0846F)
VGA Card	Onboard Intel® Ivy Bridge HD Graphics
VGA Driver	Intel® Ivy Bridge HD Graphics Ver:9.17.10.2753
LAN Card	Intel® 82579LM/82593V Gigabit Network Connection
LAN Driver	Intel® 82579LM/82593V Gigabit Network Connection Ver: 11.14.32.0/11.11.43.0
Audio Card	Realtek ALC886 High Definition Audio
Audio Driver	Realtek ALC886 High Definition Audio Ver:6.0.1.6649
Chip Driver	Intel® CHIPSET Device Software Ver:9.3.0.1019
USB 3.0 Driver	Intel® USB3.0 eXtensible Host Controll 1.0.4.225
SCSI Card	Adaptec 29160LP
SCSI HDD	Seagate ST39173W 20G
SATA HDD	Seagate ST3160815AS
Compact Flash	Apacer AP-CF001GD4CG-NR 1G
FDD	N/A
CDROM	Pioneer KBXCN2
Power Supply	FSP-350-60GC
Back plane	N/A

- **Operating Temperature**

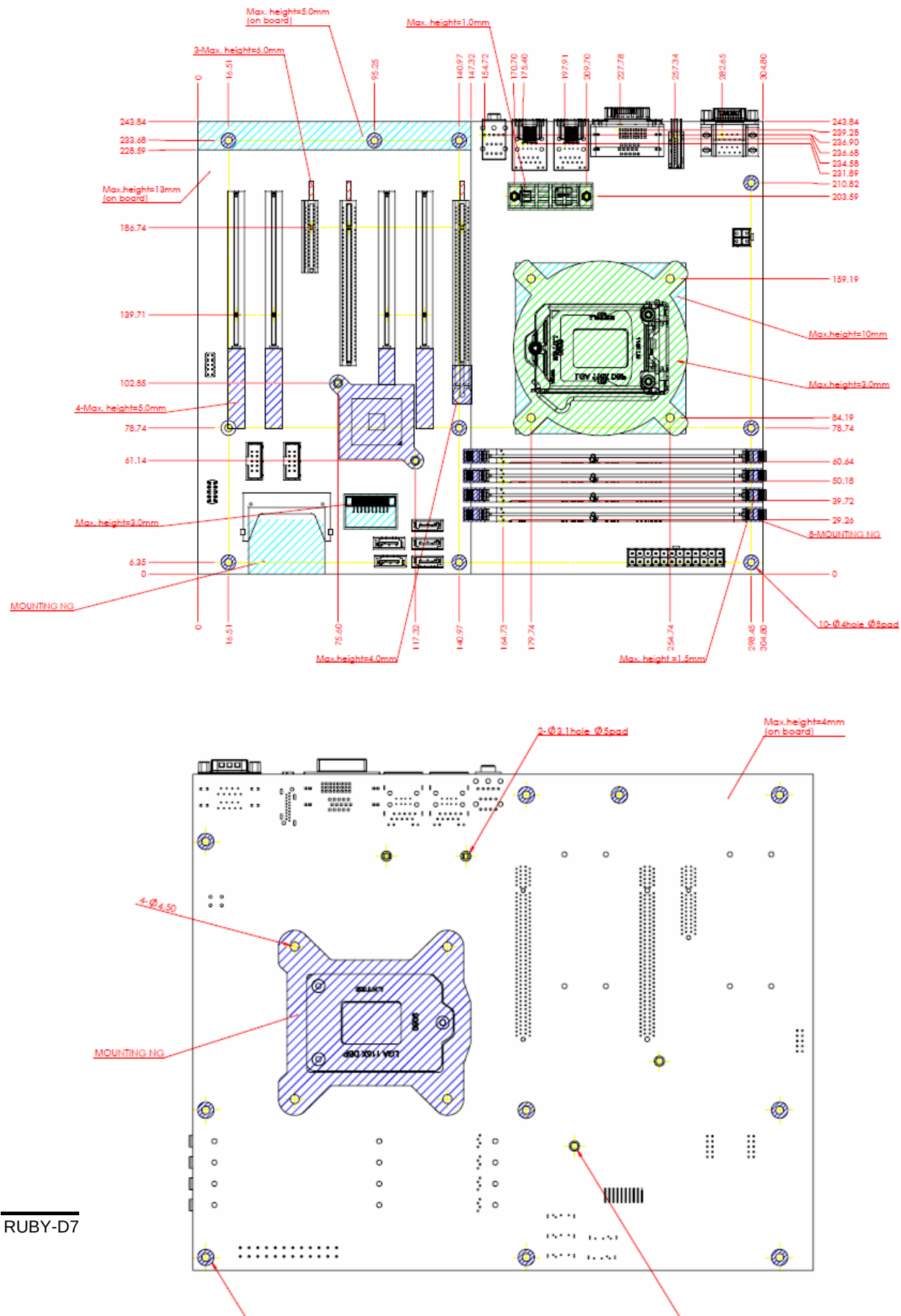
0 °C ~ 60 °C

- **Storage temperature**

-20 ~ 80 °C

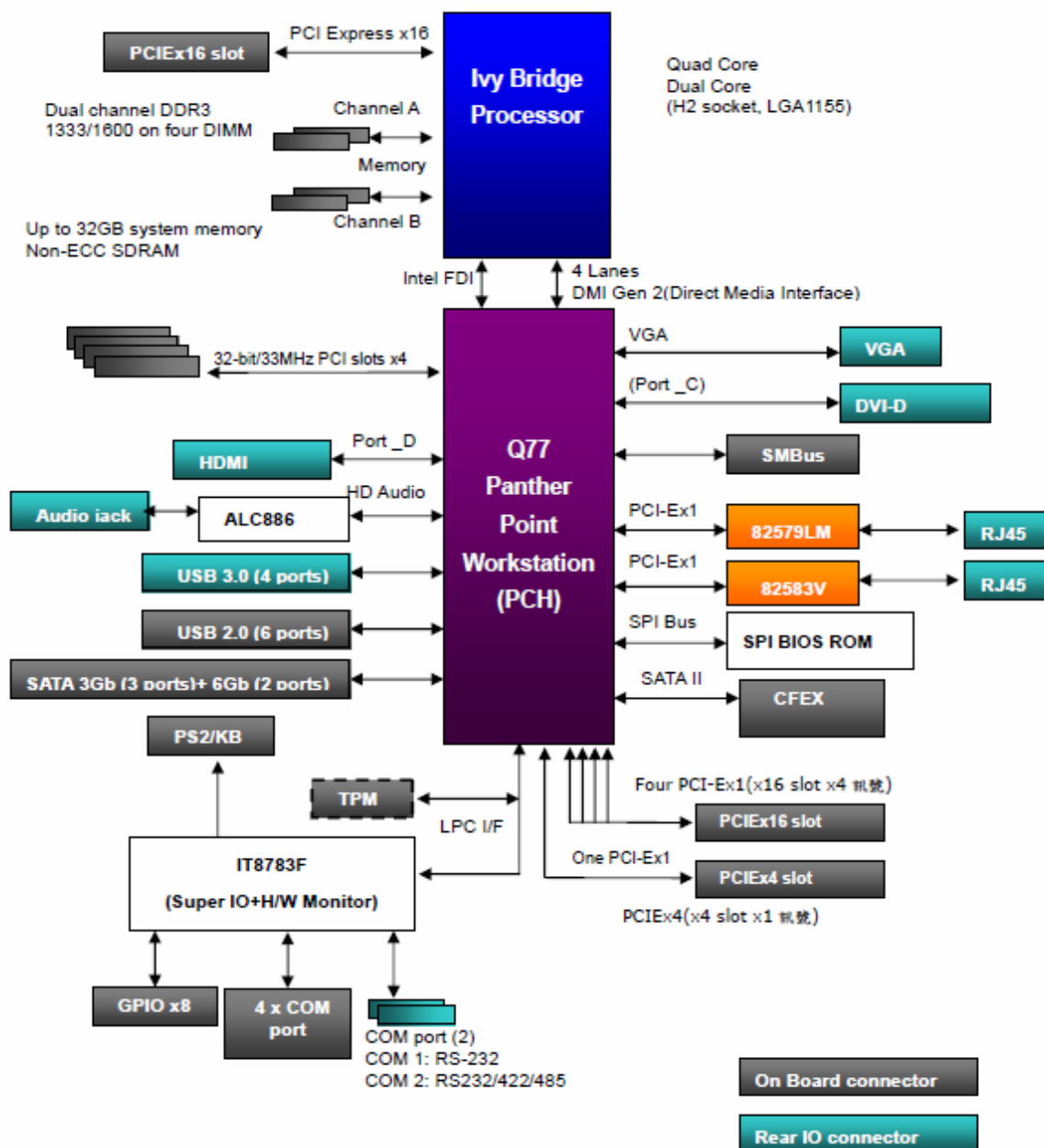
- **Relative Humidity**
0% ~ 90%, non-condensing

1.3.1 Mechanical Drawing



1.4 System Architecture

All of details operating relations are shown in RUBY-D714VG2AR System Block Diagram.



RUBY-D714VG2AR System Block Diagram

Chapter 2

Hardware Configuration

This chapter gives the definitions and shows the positions of jumpers, headers and connectors. All of the configuration jumpers on RUBY-D714VG2AR are in the proper position. The default settings shipped from factory are marked with an asterisk (★).

2.1 Jumpers

In general, jumpers on the single board computer are used to select options for certain features. Some of the jumpers are designed to be user-configurable, allowing for system enhancement. The others are for testing purpose only and should not be altered. To select any option, cover the jumper cap over (SHORT) or remove (NC) it from the jumper pins according to the following instructions. Here NC stands for “Not Connect”.

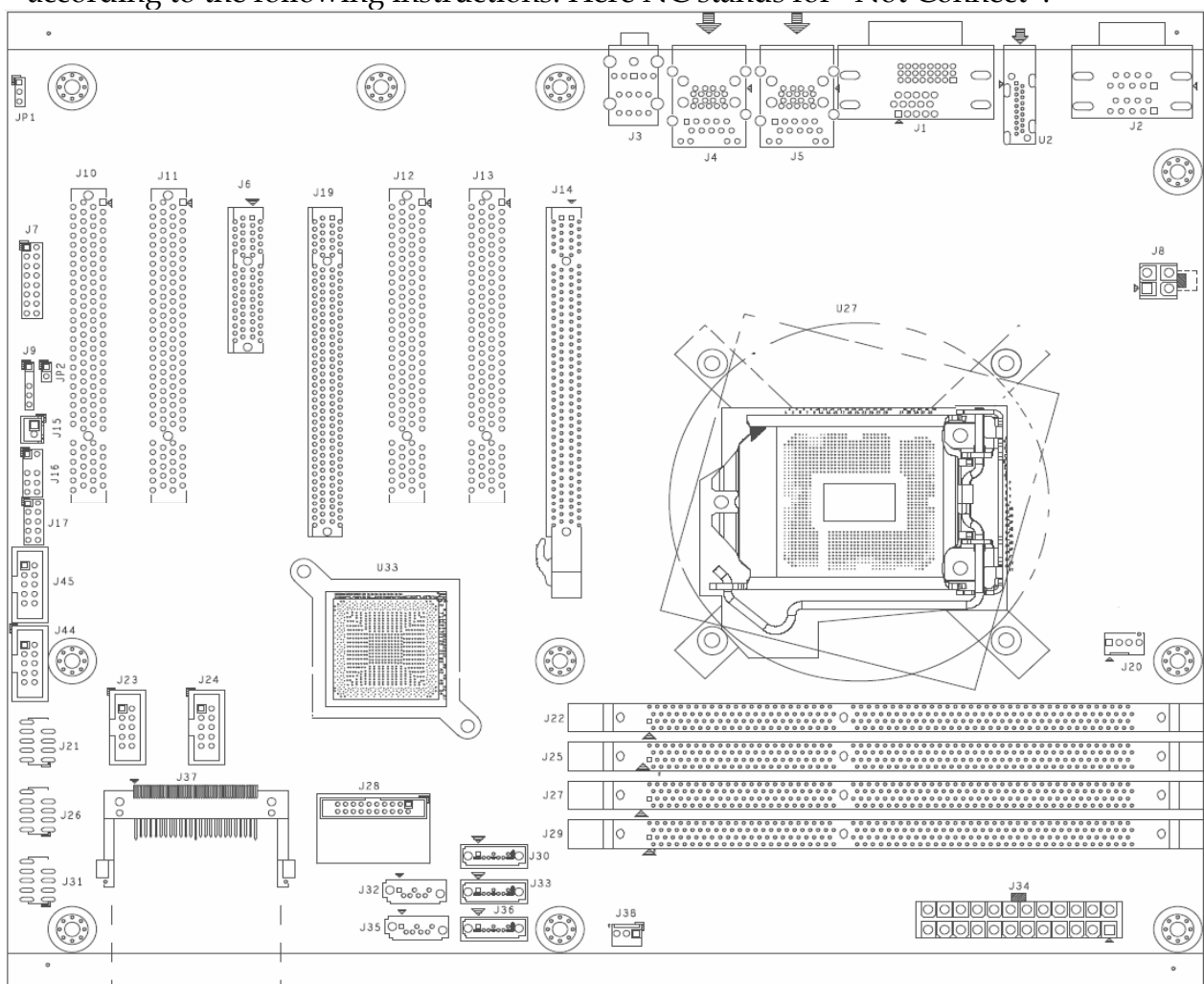


Figure 1-1 RUBY-D714VG2AR Top-side Jumper and Connector Locations

Pin Assignments of Connectors

JP1: Clear CMOS



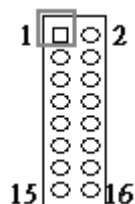
PIN No.	Signal Description
1-2	Normal
2-3	Clear CMOS

JP2: ATX/ AT Mode Select



PIN No.	Signal Description
1-2 short	ATX emulation AT mode
1-2 open	ATX mode

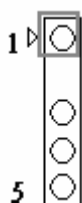
J7: Front Panel System Connector



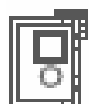
PIN No.	Signal Description	PIN No.	Signal Description
1	PWR_LED(+)	2	Speaker(+)
3	PWR_LED(-)	4	N/C
5	J4 LAN1_ACT(+)	6	N/C
7	J4 LAN1_LINK(-)	8	Speaker(-)
9	J5 LAN2_LINK(-)	10	Power Button
11	J5 LAN2_ACT(+)	12	GND
13	HDD_LED(+)	14	GND
15	HDD_LED(-)	16	Rest

J8: ATX 4Pin 12V Power Connector

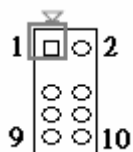
PIN No.	Signal Description
1	Ground
2	Ground
3	+12V
4	+12V

J9: SMBus Connector

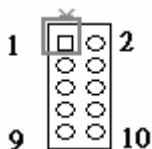
PIN No.	Signal Description
1	SMBus_CLK
2	N/C
3	Ground
4	SMBus_DAT
5	+5V

J15: Case Open Connector

PIN No.	Signal Description
1	GND
2	Case open

J16: External Keyboard/Mouse Connector

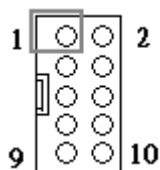
PIN No.	Signal Description	PIN No.	Signal Description
1	Mouse Data	2	Keyboard Data
3	N/C	4	N/C
5	Ground	6	Ground
7	PS2 Power	8	PS2 Power
9	Mouse Clock	10	Keyboard Clock

J17: GPIO

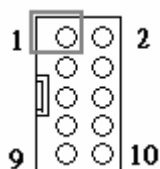
PIN No.	Signal Description	PIN No.	Signal Description
1	LPC_GP60	2	LPC_GP64
3	LPC_GP61	4	LPC_GP65
5	LPC_GP62	6	LPC_GP66
7	LPC_GP63	8	LPC_GP67
9	GND	10	Vcc

J20: CPU FAN Power Connector

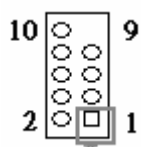
PIN No.	Signal Description
1	Ground
2	+12V
3	Fan on/off output
4	Fan Speed control

J23/24: COM3/COM4 Serial Port Connector

PIN No.	Signal Description	PIN No.	Signal Description
1	DCD (Data Carrier Detect)	2	DSR (Data Set Ready)
3	RXD (Receive Data)	4	RTS (Request to Send)
5	TXD (Transmit Data)	6	CTS (Clear to Send)
7	DTR (Data Terminal Ready)	8	RI (Ring Indicator)
9	GND (Ground)	10	N/C

J44/45: COM5/COM6 Serial Port Connector

PIN No.	Signal Description	PIN No.	Signal Description
1	DCD (Data Carrier Detect)	2	DSR (Data Set Ready)
3	RXD (Receive Data)	4	RTS (Request to Send)
5	TXD (Transmit Data)	6	CTS (Clear to Send)
7	DTR (Data Terminal Ready)	8	RI (Ring Indicator)
9	GND (Ground)	10	N/C

J21/J26/J31: External USB Connector

PIN No.	Signal Description	PIN No.	Signal Description
1	5V Dual	2	5V Dual
3	USB-	4	USB-
5	USB+	6	USB+
7	Ground	8	Ground
9			N/C

J38: SYSTEM FAN Power Connector

PIN No.	Signal Description
1	Ground
2	Fan speed control
3	Fan on/off output

2.2 Connector Allocation

I/O peripheral devices are connected to the interface connectors.

Connector Function List

Connector	Function	Remark
J1	DVI & VGA Port	
J2	COM PORT	COM2 Support 232/422/485
J3	Audio connector	
J4	USB 0&1/100+Giga Lan Connector (LAN1) 82579LM	
J5	USB 2&3/100+Giga Lan Connector (LAN2) 82583V	
J6	PCI-E x4 Slot	only Support x1 signal
J7	Front Panel System Connector	
J8	ATX 4Pin 12V Power Connector	
J9	SMBUS HEADER	
J10、J11、J12、J13	PCI Slot	
J14	PCI-E x16 Slot	Support PCI-E Gen3
J15	Case Open Connector	
J16	External Keyboard/Mouse Connector	
J17	8Bit GPIO Connector	
J19	PCI-E x16 Slot	only Support x4 signal
J20	CPU FAN Power Connector	
J21、J26、J31	External USB Connector	
J22、J25、J27、J29	DIMM	
J23、J24、J44、J45	COM3/COM4/COM5/COM6 Serial Port Connector	
J28	TPM(Trusted Platform Module) Connector	

J30、J33、J36	SATA Connector(3Gb/s)	
J32、J35	SATA Connector(6Gb/s)	
J34	ATX Power	
J37	CFEX Slot	
J38	System FAN Power Connector	

Chapter 3

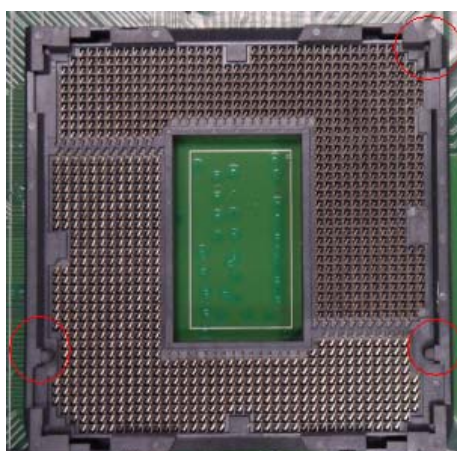
System Installation

This chapter provides the instructions to set up the system. The additional information is enclosed to help you set up onboard devices

3.1 Intel® Dual Core/Quad Core processor

LGA-1155 CPU Socket

Pin1 corner of the CPU Socket



Alignment key

LGA-1155 CPU

Yellow Triangle Pin1 of the CPU



Notch

Please remember to locate the alignment keys on the CPU socket of the motherboard and the notches on the CPU.

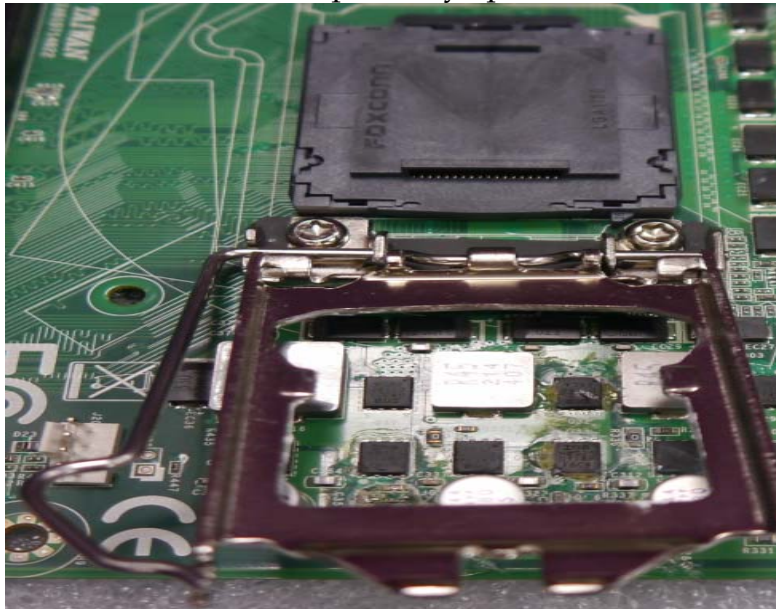
LGA-1155 CPU Installation Steps

Before install the CPU, please make sure to turn off the power first!!

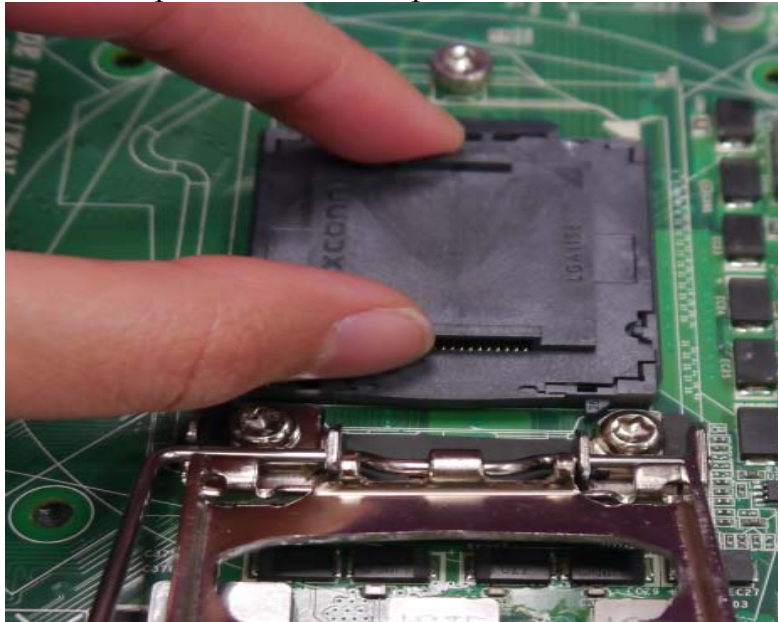
1. Open the load lever.



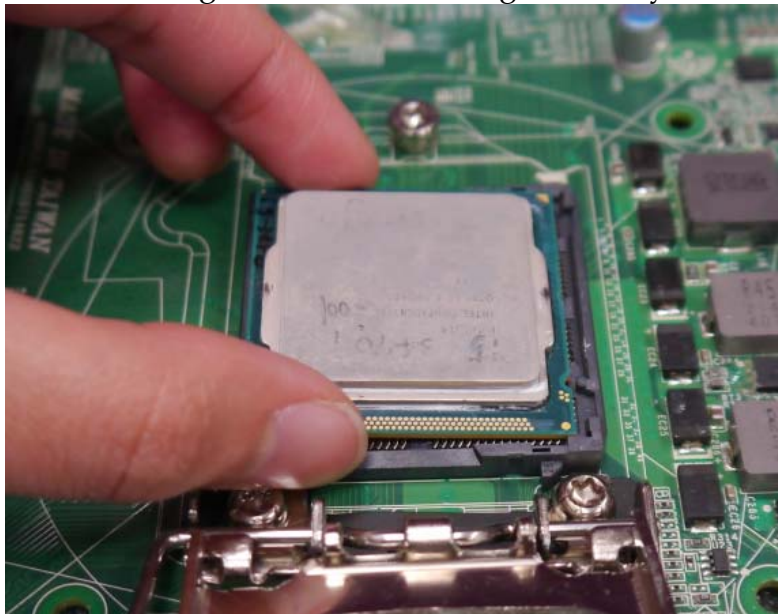
2. Lift the load lever up to fully open



3. Remove the plastic cap on the CPU socket. Before you install the CPU, always cover it to protect the socket pin.



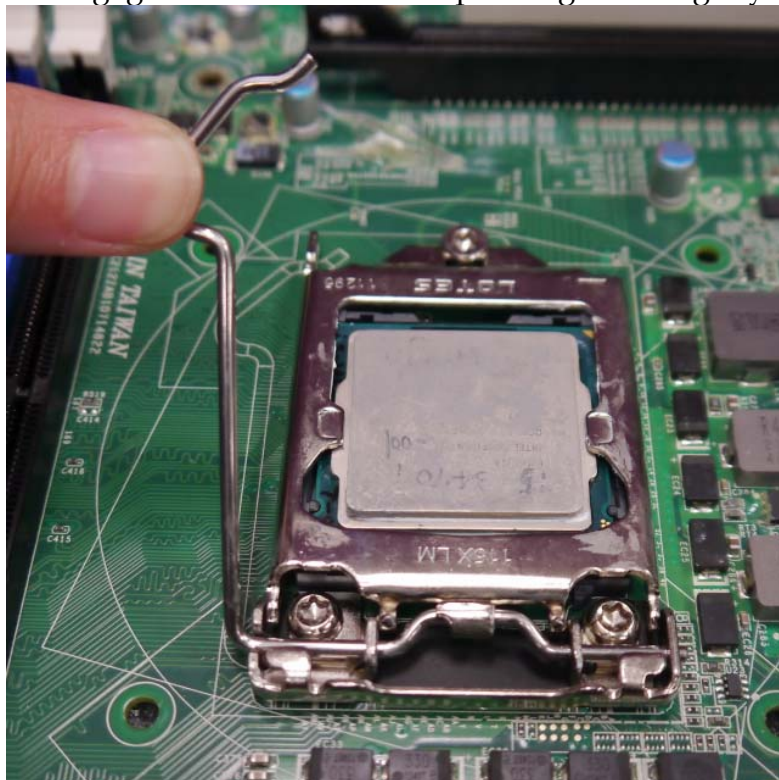
4. After confirming the CPU direction for correct mating, put down the CPU in the socket housing frame. Note that alignment keys are matched.



5. Make sure the CPU has been seated well into the socket. If not, take out the CPU and reinstall.



6. Engage the load lever while pressing down lightly onto the load plate.



7. Push the CPU socket lever back into its locked position.



8. Please make sure four hooks are in proper position before you install the core.

3.2 Main Memory

RUBY-D714 provide 4 x240 pin DIMM sockets (Dual Channel) which supports Dual channel 1066/1333 DDR3-SDRAM as main memory, non-register function. RUBY-D714 just supports Non-ECC memory. The maximum memory can be up to 32GB. Memory clock and related settings can be detected by BIOS via SPD interface.

For system compatibility and stability, do not use memory module without brand. Memory configuration can be set to either one double-sided DIMM in one DIMM socket or two single-sided DIMM in both sockets.

Beware of the connection and lock integrity from memory module to socket. Inserting improperly it will affect the system reliability.

Before locking, make sure that all modules have been fully inserted into the card slots

Note:

To insure the system stability, please do not change any of DRAM parameters in BIOS setup to modify system the performance without acquired technical information.

3.3 Installing the Single Board Computer

To install your RUBY-D714 into standard chassis or proprietary environment, please perform the following:

Step 1 : Check all jumpers setting on proper position

Step 2 : Install and configure CPU and memory module at right position

Step 3 : Place RUBY-D714 into the dedicated position in the system

Step 4 : Attach cables to existing peripheral devices and secure it

WARNING

Please ensure that SBC is properly inserted and fixed by mechanism.

Note

Please refer to section 3.3.1 to 3.3.7 to install INF/VGA/LAN/Audio drivers.

3.3.1 Chipset Component Driver

RUBY-D714 uses state-of-art Intel® Panther Point chipset. It's a new chipset that some old operating systems might not be able to recognize. To overcome this compatibility issue, for previous Windows Operating Systems such as Windows XP, please install its INF before any of other Drivers are installed. You can find very easily this chipset component driver in RUBY-D714 CD-title.

Moreover, if using some old OS, the driver may not be supported anymore. We recommend to change the different OS to comply with this new chipset.

3.3.2 Intel® HD Graphics Family

With latest Intel® Core Ivy Bridge series structure, RUBY-D714 Intel® HD Graphic is built in with CPU. Therefore 2nd Generation Core and 3rd Generation Core CPUs provide HD integrated Graphic support sharing on board physical memories. RUBY-D714 has both internal VGA & DVI interface. This combination makes RUBY-D714 an excellent piece of multimedia hardware

With no additional video adaptor, this onboard video will usually be the system display output. By adjusting the BIOS setting to disable on-board VGA, an add-on PCI-Express Graphic card can take over the system display.

※To use Intel® Integrated HD Graphic, it's required to choose the CPU which has Integrated Graphic built-in. Otherwise there will be no display whatsoever.

3.3.3 Intel® PROSet Gigabit Ethernet Controlle

Drivers Support

Please find Intel® 82583V and 82579LM LAN driver in /Ethernet directory of RUBY-D714 CD-title. The driver supports Windows Win7 64-bits.

3.3.4 Audio Controller

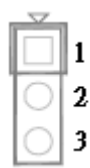
Please find Intel® High Definition Audio driver from RUBY-D714 CD-title. The driver supports Windows Win7 64-bits.

3.3.5 Intel® Active Management Technology (Intel® AMT)

Please find the latest Intel®AMT 7.0 driver from RUBY-D714 CD-title. The driver supports Windows Win7 64-bits.

3.4 Clear CMOS Operation

JP1: CLEAR CMOS



PIN No.	Signal Description
1-2 Short	Normal Operation
2-3 Short	Clear CMOS Contents

3.5 WDT Function

The Watchdog Timer of motherboard consists of 8-bit programmable time-out counter and a control and status register.

WDT Controller Register

There are two PNP I/O port addresses that can be used to configure WDT.

2Eh: EFIR (Extended Function Index Register, for identifying CR index number)

2Fh: EFDR (Extended Function Data Register, for accessing desired CR)

WDT Control Mode Register

The working algorithm of the WDT function can be simply described as a counting process. The Time-Out Interval can be set through software programming. The availability of the time-out interval settings by software or hardware varies from boards to boards.

RUBY-D714 allows users to control WDT through dynamic software programming. The WDT starts counting when it is activated. It sends out a signal to system reset or to non-maskable interrupt (NMI), when time-out interval ends. To prevent the time-out interval from running out, a re-trigger signal will need to be sent before the counting reaches its end. This action will restart the counting process. A well-written WDT program should keep the counting process running under normal condition. WDT should never generate a system reset or NMI signal unless the system runs into troubles.

The related Control Registers of WDT are all included in the following sample program that is written in C language. User can fill a non-zero value into the Time-out Value Register to enable/refresh WDT. System will be reset after the Time-out Value to be counted down to zero. Or user can directly fill a zero value into Time-out Value Register to disable WDT immediately. To ensure a successful accessing to the content of desired Control Register, the sequence of following program codes should be step-by-step run again when each register is accessed.

Additionally, there are maximum 2 seconds of counting tolerance that should be considered into user' application program. For more information about WDT, please refer to ITE IT8783F data sheet.

There are two PNP I/O port addresses that can be used to configure WDT,
1) 0x2E:EFIR (Extended Function Index Register, for identifying CR index number)
2) 0x2F:EFDR (Extended Function Data Register, for accessing desired CR)

WDT Control Command Example

```
#include <stdio.h>
#include <conio.h>
#include <dos.h>

#define SIO_Port 0x2E
#define SIO_Port2 0x4E
#define GPIO_LDN 0x07

void Enter_IT8783_SIO() {
    outportb(SIO_Port, 0x87);
    outportb(SIO_Port, 0x01);
    outportb(SIO_Port, 0x55);
    outportb(SIO_Port, 0x55);
}

void Set_LDN(unsigned char LDN) {
    outportb(SIO_Port, 0x07);
    outportb(SIO_Port+1, LDN);
    printf("LDN=%x\n", LDN);
}
```

```
}

void Set_Register(unsigned char offset, unsigned char value) {
    outportb(SIO_Port, offset);
    outportb(SIO_Port+1, value);
    printf("Write offset:%x = %x\n", offset, value);
}

int main(void) {

    printf("test string\n");
    Enter_IT8783_SIO();

    Set_LDN(GPIO_LDN);

    Set_Register(0x72, 0xC0);
    Set_Register(0x73, 0x05);
    printf("System will reset in 5 seconds\n");

    return 0;
}
```

3.6 GPIO

The motherboard provides 4 input / output ports that can be individually configured to perform a simple basic I/O function.

GPIO Pin Assignment

The RUBY-D714 provides 4 input/output ports that can be individually configured to perform a simple basic I/O function. Users can configure each individual port to become an input or output port by programming register bit of I/O Selection. To invert port value, the setting of Inversion Register has to be made. Port values can be set to read or write through Data Register.

The GPIO port is located on J17 shown as follows. Please note: Do not short the Pin 9 and Pin 10 of the **J17**!

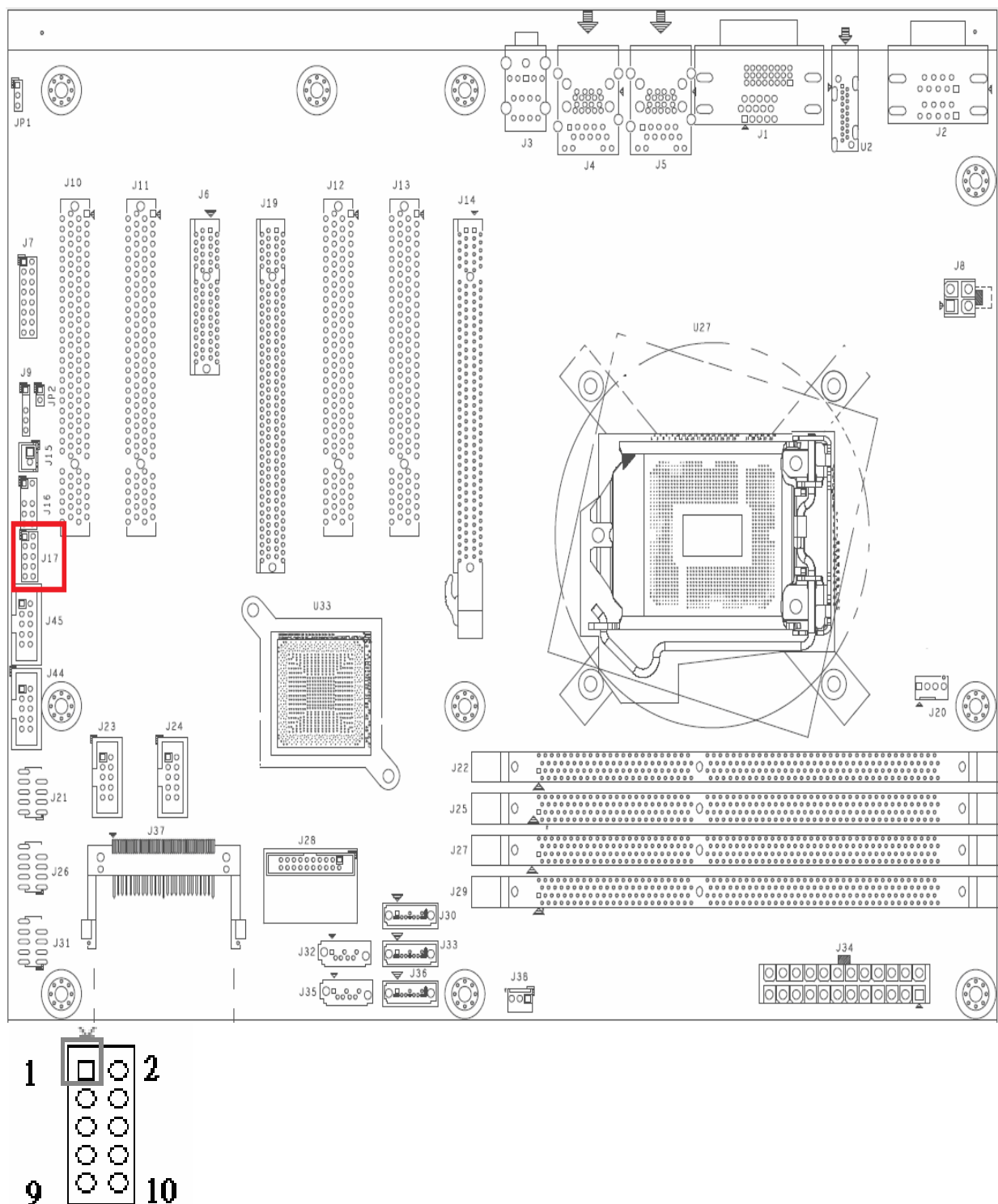
The control for the GPIO signals is handled through a separate 1-byte I/O space.

J17: General Purpose I/O Connector

PIN No.	Signal Description	PIN No.	Signal Description
1	LPC_GP60	2	LPC_GP64
3	LPC_GP61	4	LPC_GP65
5	LPC_GP62	6	LPC_GP66
7	LPC_GP63	8	LPC_GP67
9	GND	10	Vcc

Note:

All General Purpose I/O ports can only apply to standard TTL $\pm 5\%$ signal level (0V/5V), and each Fan.



GPIO Control Command Example (C Language)

```
/* This File was created by GPIO APP,Please don't EDIT*/
/*****
Sample Code for ITE series SIO GPIO Access

SIO IO INDEX : 0x2E
GPIO IO BASE : 0x0A00
GPIO PIN Define:
Pin 1 : GP60
Pin 2 : GP61
Pin 3 : GP62
Pin 4 : GP63
Pin 5 : GP64
Pin 6 : GP65
Pin 7 : GP66
Pin 8 : GP67
*****/

#include <stdio.h>
#include <conio.h>
#include <string.h>
#include <dos.h>

/*****
Common defines
*****/

#define HIGH 1
#define LOW 0
#define IN 1
#define OUT 0

/*=====

Prototype defines
=====*/

int Get_IT8783_GPIOBase(
    int IOINDEX
);

void Enter_IT8783_SIO(
    int IOINDEX
);

void Exit_IT8783_SIO(
    int IOINDEX
```

```
);

void Set_IT8783_LDN(
    int IOINDEX,
    int LDN
);

int Get_IT8783_Byte_Val(
    int IOINDEX,
    int offset
);

void Set_IT8783_Byte_Val(
    int IOINDEX,
    int offset,
    int Data
);

/*****
MAIN Code Start
*****/
int main()
{
    return;
}

/*****
Procedure Name : Get Super IO GPIO Base address
Check defines in config file name

input 1 : INT IOINDEX
return   : GPIO BASE
*****/
int Get_IT8783_GPIOBase(
    int IOINDEX
)
{
    int IODATA = IOINDEX + 1;
    int GPIOBASE;

    //SIO Enter Key
    Enter_IT8783_SIO(IOINDEX);
    //Change LDN to 7 (GPIO)
    Set_IT8783_LDN(IOINDEX,0x07);
    //Get GPIO BASE
    GPIOBASE = Get_IT8783_Byte_Val(IOINDEX,0x62) * 0x100;
```

```
GPIOBASE = GPIOBASE + Get_IT8783_Byte_Val(IOINDEX,0x63);
return GPIOBASE;
}
```

```
/*****
```

Procedure Name : Enter IT8783 Super IO

Check defines in config file name

input 1 : INT IOINDEX

```
*****/
```

```
void Enter_IT8783_SIO(
```

```
    int IOINDEX
```

```
)
```

```
{
```

```
    int IODATA = IOINDEX + 1;
```

```
    outportb(IOINDEX,0x87);
```

```
    outportb(IOINDEX,0x01);
```

```
    outportb(IOINDEX,0x55);
```

```
    outportb(IOINDEX,0x55);
```

```
    return;
```

```
}
```

```
/*****
```

Procedure Name : Exit IT8783 Super IO

Check defines in config file name

input 1 : INT IOINDEX

```
*****/
```

```
void Exit_IT8783_SIO(
```

```
    int IOINDEX
```

```
)
```

```
{
```

```
    int IODATA = IOINDEX + 1;
```

```
    outportb(IOINDEX,0x02);
```

```
    outportb(IODATA,0x02);
```

```
    return;
```

```
}
```

```
/*****
```

Procedure Name : Change IT8783 LDN

Description : Change Super IO Logical Device Number

input 1 : INT IOINDEX

input 2 : INT LDN(Logical Device Number)

```
***** /
```

```
void Set_IT8783_LDN(  
    int IOINDEX,  
    int LDN  
)  
{  
int IODATA = IOINDEX + 1;  
  
outportb(IOINDEX,0x07);  
outportb(IODATA, LDN);  
return;  
}
```

```
/*****
```

Procedure Name : Get IT8783 offset data

Description : Get IT8783 offset byte data

input 1 : INT IOINDEX
input 2 : INT offset
return : INT Data (Byte)

```
***** /
```

```
int Get_IT8783_Byte_Val(  
    int IOINDEX,  
    int offset  
)  
{  
int Data;  
int IODATA = IOINDEX + 1;
```

```
outportb(IOINDEX,offset);           //set offset of data read  
Data = inportb(IODATA);              //Get Data from offset  
return Data;  
}
```

```
/*****
```

Procedure Name : Set IT8783 offset data

Description : Set Byte data to IT8783 offset

input 1 : INT IOINDEX
input 2 : INT offset
input 3 : INT Data

```
***** /
```

```
void Set_IT8783_Byte_Val(  
    int IOINDEX,  
    int offset,  
    int Data
```

```
)  
{  
int IODATA = IOINDEX + 1;
```

```
outportb(IOINDEX,offset);  
outportb(IODATA, Data);  
return;  
}
```

```
/*****
```

Procedure Name : Set GPIO Pin 1 Mode & In/Out data

Description : Set GPIO Pin I/O Mode

if GPIOMode is Input , input 2 data will
dont care. and this function will return
read value.

Pin 1 : GP60

input 1 : INT GPIOMode

input 2 : INT Data

Out 1 : INT Data

```
*****/
```

```
int SET_GPIO_PIN1(  
int GPIOMode,  
int Data  
)  
{
```

```
    //GPIO Mode is Output
```

```
    if(GPIOMode = 0){
```

```
        Set_IT8783_Byte_Val(0x2E,0xCD,
```

```
Get_IT8783_Byte_Val(0x2E,0xCD) | (1<<0));
```

```
        //if Data is HIGH
```

```
        if(Data==1){
```

```
            outportb(0x0A00 + 5, 1<<0 );
```

```
        }
```

```
        //if Data is LOW
```

```
        if(Data==0){
```

```
            outportb(0x0A00 + 5, (inportb(0x0A00 + 5) & ~(1<<0) ) );
```

```
        }
```

```
        return NULL;
```

```
    }
```

```
    //GPIO Mode is Input
```

```
    else if(GPIOMode = 1){
```

```
        Set_IT8783_Byte_Val(0x2E,0xCD,
```

```
Get_IT8783_Byte_Val(0x2E,0xCD) & ~(1<<0));
```

```
        Data = inportb(0x0A00 + 5) & (1<<0);
        return    Data;
    }

    return NULL;
}
```

/*****

Procedure Name : Set GPIO Pin 2 Mode & In/Out data

Description : Set GPIO Pin I/O Mode

if GPIOMode is Input , input 2 data will
dont care. and this function will return
read value.

Pin 2 : GP61

input 1 : INT GPIOMode

input 2 : INT Data

Out 1 : INT Data

*****/

```
int SET_GPIO_PIN2(
int GPIOMode,
int Data
)
{
    //GPIO Mode is Output
    if(GPIOMode = 0){
        Set_IT8783_Byte_Val(0x2E,0xCD,
Get_IT8783_Byte_Val(0x2E,0xCD) | (1<<1));

        //if Data is HIGH
        if(Data==1){
            outportb(0x0A00 + 5, 1<<1 );
        }
        //if Data is LOW
```

```
        if(Data==0){
            outportb(0x0A00 + 5, (inportb(0x0A00 + 5) & ~(1<<1) ) );
        }
        return NULL;
    }
    //GPIO Mode is Input
    else if(GPIOMode = 1){
        Set_IT8783_Byte_Val(0x2E,0xCD,
Get_IT8783_Byte_Val(0x2E,0xCD) & ~(1<<1));

        Data = inportb(0x0A00 + 5) & (1<<1);
        return    Data;
    }

    return NULL;
}

/*****
```

Procedure Name : Set GPIO Pin 3 Mode & In/Out data

Description : Set GPIO Pin I/O Mode

if GPIOMode is Input , input 2 data will
dont care. and this function will return
read value.

Pin 3 : GP62

input 1 : INT GPIOMode

input 2 : INT Data

Out 1 : INT Data

*****/

```
int SET_GPIO_PIN3(
int GPIOMode,
int Data
)
{
```

```
//GPIO Mode is Output
if(GPIOMode = 0){
    Set_IT8783_Byte_Val(0x2E,0xCD,
Get_IT8783_Byte_Val(0x2E,0xCD) | (1<<2));

    //if Data is HIGH
    if(Data==1){
        outportb(0x0A00 + 5, 1<<2 );
    }
    //if Data is LOW
    if(Data==0){
        outportb(0x0A00 + 5, (inportb(0x0A00 + 5) & ~(1<<2) ) );
    }
    return NULL;
}
//GPIO Mode is Input
else if(GPIOMode = 1){
    Set_IT8783_Byte_Val(0x2E,0xCD,
Get_IT8783_Byte_Val(0x2E,0xCD) & ~(1<<2));

    Data = inportb(0x0A00 + 5) & (1<<2);
    return    Data;
}

return NULL;
}

/*****
```

Procedure Name : Set GPIO Pin 4 Mode & In/Out data

Description : Set GPIO Pin I/O Mode

if GPIOMode is Input , input 2 data will
dont care. and this function will return
read value.

Pin 4 : GP63

input 1 : INT GPIOMode

input 2 : INT Data

Out 1 : INT Data

```
*****/  
  
int SET_GPIO_PIN4(  
int GPIOMode,  
int Data  
)  
{  
    //GPIO Mode is Output  
    if(GPIOMode = 0){  
        Set_IT8783_Byte_Val(0x2E,0xCD,  
Get_IT8783_Byte_Val(0x2E,0xCD) | (1<<3));  
  
        //if Data is HIGH  
        if(Data==1){  
            outputb(0x0A00 + 5, 1<<3 );  
        }  
        //if Data is LOW  
        if(Data==0){  
            outputb(0x0A00 + 5, (inportb(0x0A00 + 5) & ~(1<<3) ) );  
        }  
        return NULL;  
    }  
    //GPIO Mode is Input  
    else if(GPIOMode = 1){  
        Set_IT8783_Byte_Val(0x2E,0xCD,  
Get_IT8783_Byte_Val(0x2E,0xCD) & ~(1<<3));  
  
        Data = inportb(0x0A00 + 5) & (1<<3);  
        return Data;  
    }  
    return NULL;  
}  
  
/*****
```

Procedure Name : Set GPIO Pin 5 Mode & In/Out data

Description : Set GPIO Pin I/O Mode

if GPIOMode is Input , input 2 data will

dont care. and this function will return

read value.

Pin 5 : GP64

input 1 : INT GPIOMode

input 2 : INT Data

Out 1 : INT Data

```
*****/

int SET_GPIO_PIN5(
int GPIOMode,
int Data
)
{
    //GPIO Mode is Output
    if(GPIOMode = 0){
        Set_IT8783_Byte_Val(0x2E,0xCD,
Get_IT8783_Byte_Val(0x2E,0xCD) | (1<<4));

        //if Data is HIGH
        if(Data==1){
            outputb(0x0A00 + 5, 1<<4 );
        }
        //if Data is LOW
        if(Data==0){
            outputb(0x0A00 + 5, (inportb(0x0A00 + 5) & ~(1<<4) ) );
        }
        return NULL;
    }
    //GPIO Mode is Input
    else if(GPIOMode = 1){
        Set_IT8783_Byte_Val(0x2E,0xCD,
Get_IT8783_Byte_Val(0x2E,0xCD) & ~(1<<4));

        Data = inportb(0x0A00 + 5) & (1<<4);
        return Data;
    }
    return NULL;
}
```

```
/*****
```

Procedure Name : Set GPIO Pin 6 Mode & In/Out data

Description : Set GPIO Pin I/O Mode

if GPIOMode is Input , input 2 data will
dont care. and this function will return
read value.

Pin 6 : GP65

input 1 : INT GPIOMode

input 2 : INT Data

Out 1 : INT Data

```
*****/
```

```
int SET_GPIO_PIN6(  
int GPIOMode,  
int Data  
)  
{  
    //GPIO Mode is Output  
    if(GPIOMode == 0){  
        Set_IT8783_Byte_Val(0x2E,0xCD,  
Get_IT8783_Byte_Val(0x2E,0xCD) | (1<<5));  
  
        //if Data is HIGH  
        if(Data==1){  
            outputb(0x0A00 + 5, 1<<5 );  
        }  
        //if Data is LOW  
        if(Data==0){  
            outputb(0x0A00 + 5, (inportb(0x0A00 + 5) & ~(1<<5) ) );  
        }  
        return NULL;  
    }  
    //GPIO Mode is Input
```

```
        else if(GPIOMode = 1){
            Set_IT8783_Byte_Val(0x2E,0xCD,
Get_IT8783_Byte_Val(0x2E,0xCD) & ~(1<<5));

            Data = inportb(0x0A00 + 5) & (1<<5);
            return Data;
        }
        return NULL;
    }

/*****
```

Procedure Name : Set GPIO Pin 7 Mode & In/Out data

Description : Set GPIO Pin I/O Mode

if GPIOMode is Input , input 2 data will
dont care. and this function will return
read value.

Pin 7 : GP66

input 1 : INT GPIOMode

input 2 : INT Data

Out 1 : INT Data

```
*****/

int SET_GPIO_PIN7(
int GPIOMode,
int Data
)
{
    //GPIO Mode is Output
    if(GPIOMode = 0){
        Set_IT8783_Byte_Val(0x2E,0xCD,
Get_IT8783_Byte_Val(0x2E,0xCD) | (1<<6));

        //if Data is HIGH
        if(Data==1){
```

```
        outportb(0x0A00 + 5, 1<<6 );
    }
    //if Data is LOW
    if(Data==0){
        outportb(0x0A00 + 5, (inportb(0x0A00 + 5) & ~(1<<6) ) );
    }
    return NULL;
}
//GPIO Mode is Input
else if(GPIOMode = 1){
    Set_IT8783_Byte_Val(0x2E,0xCD,
Get_IT8783_Byte_Val(0x2E,0xCD) & ~(1<<6));

        Data = inportb(0x0A00 + 5) & (1<<6);
        return    Data;
    }
    return NULL;
}

/*****
```

Procedure Name : Set GPIO Pin 8 Mode & In/Out data

Description : Set GPIO Pin I/O Mode

if GPIOMode is Input , input 2 data will
dont care. and this function will return
read value.

Pin 8 : GP67

input 1 : INT GPIOMode

input 2 : INT Data

Out 1 : INT Data

*****/

```
int SET_GPIO_PIN8(
int GPIOMode,
int Data
```

```
)  
{  
    //GPIO Mode is Output  
    if(GPIOMode = 0){  
        Set_IT8783_Byte_Val(0x2E,0xCD,  
Get_IT8783_Byte_Val(0x2E,0xCD) | (1<<7));  
  
        //if Data is HIGH  
        if(Data==1){  
            outportb(0x0A00 + 5, 1<<7 );  
        }  
        //if Data is LOW  
        if(Data==0){  
            outportb(0x0A00 + 5, (inportb(0x0A00 + 5) & ~(1<<7) ) );  
        }  
        return NULL;  
    }  
    //GPIO Mode is Input  
    else if(GPIOMode = 1){  
        Set_IT8783_Byte_Val(0x2E,0xCD,  
Get_IT8783_Byte_Val(0x2E,0xCD) & ~(1<<7));  
  
        Data = inportb(0x0A00 + 5) & (1<<7);  
        return    Data;  
    }  
    return NULL;  
}  
/* This File was created by GPIO APP,Please don`t EDIT*/
```

Chapter 4

BIOS Setup Information

RUBY-D714 is equipped with the Phoenix BIOS stored in Flash ROM. These BIOS has a built-in Setup program that allows users to modify the basic system configuration easily. This type of information is stored in CMOS RAM so that it is retained during power-off periods. When system is turned on, RUBY-D714 communicates with peripheral devices and checks its hardware resources against the configuration information stored in the CMOS memory. If any error is detected, or the CMOS parameters need to be initially defined, the diagnostic program will prompt the user to enter the SETUP program. Some errors are significant enough to abort the start up.

4.1 Entering Setup -- Launch System Setup

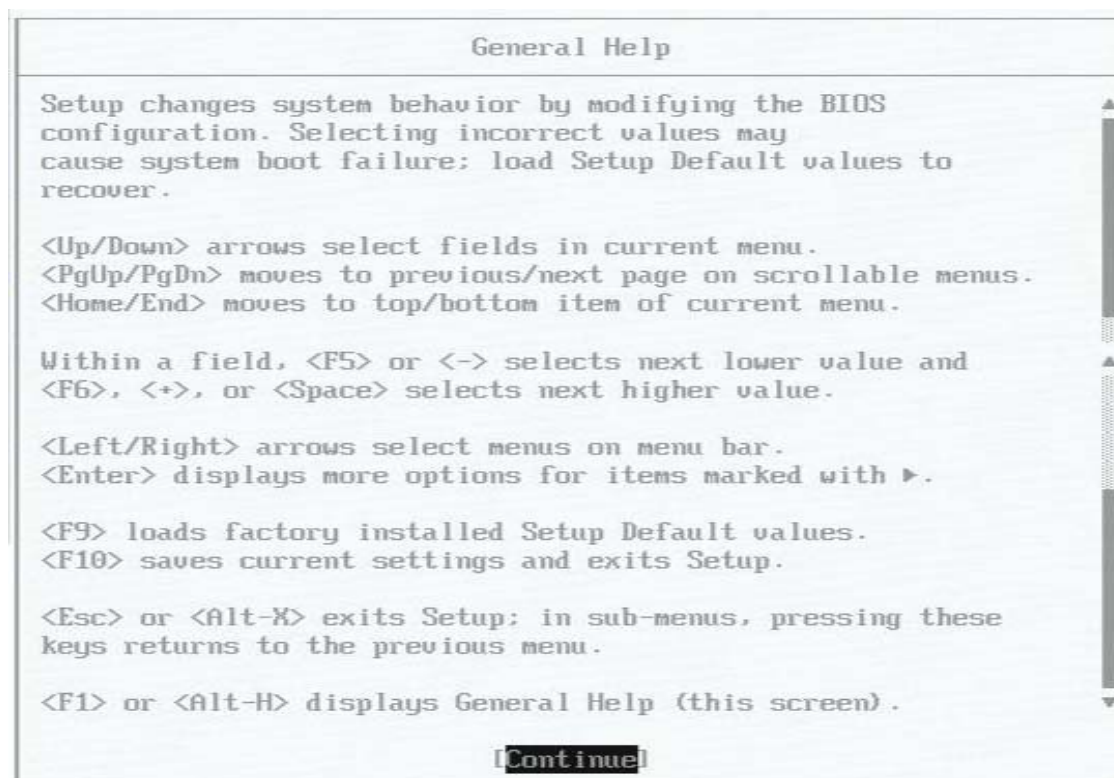
Power on the computer and the system will start POST (Power On Self Test) process. When the message below appears on the screen, press <F2> key will enter BIOS setup screen.

Press <F2> to enter SETUP

If the message disappears before responding and still wish to enter Setup, please restart the system by turning it OFF and On or pressing the RESET button. It can be also restarted by pressing <Ctrl>, <Alt>, and <Delete> keys on keyboard simultaneously.

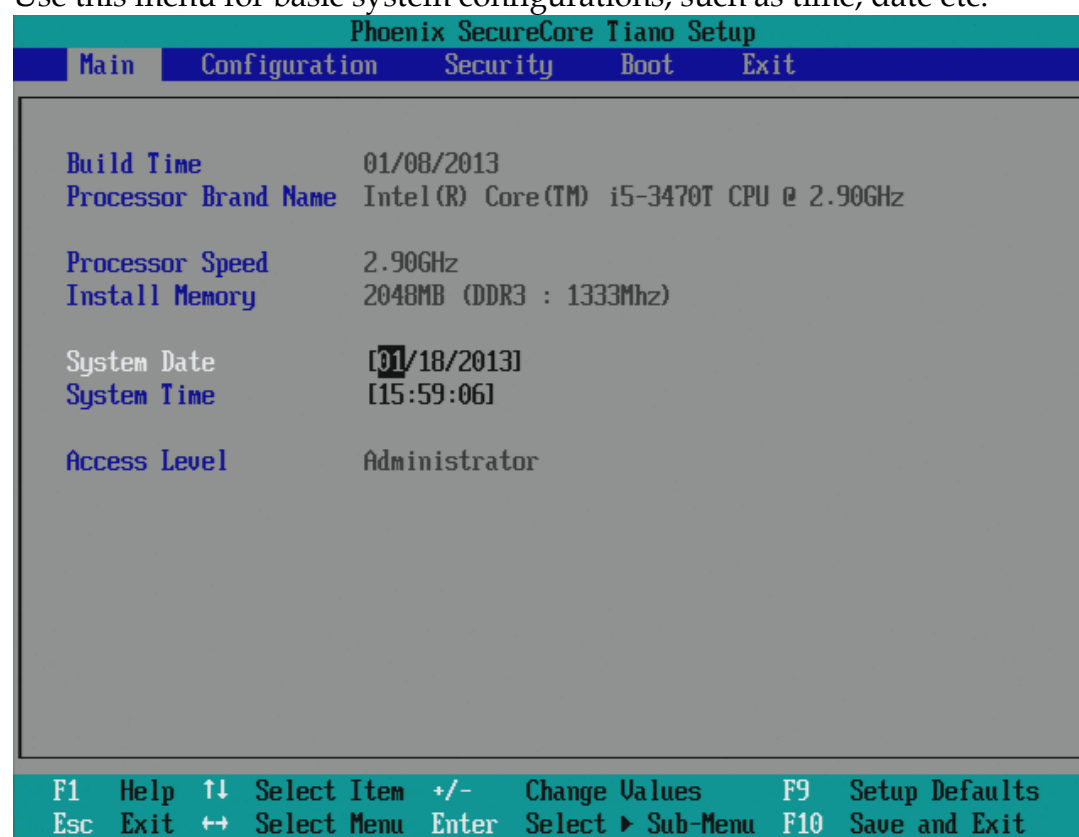
Press <F1> to Run General Help or Resume

The BIOS setup program provides a General Help screen. The menu can be easily called up from any menu by pressing <F1>. The Help screen lists all the possible keys to use and the selections for the highlighted item. Press <Esc> to exit the Help screen.



4.2 Main

Use this menu for basic system configurations, such as time, date etc.



System Date

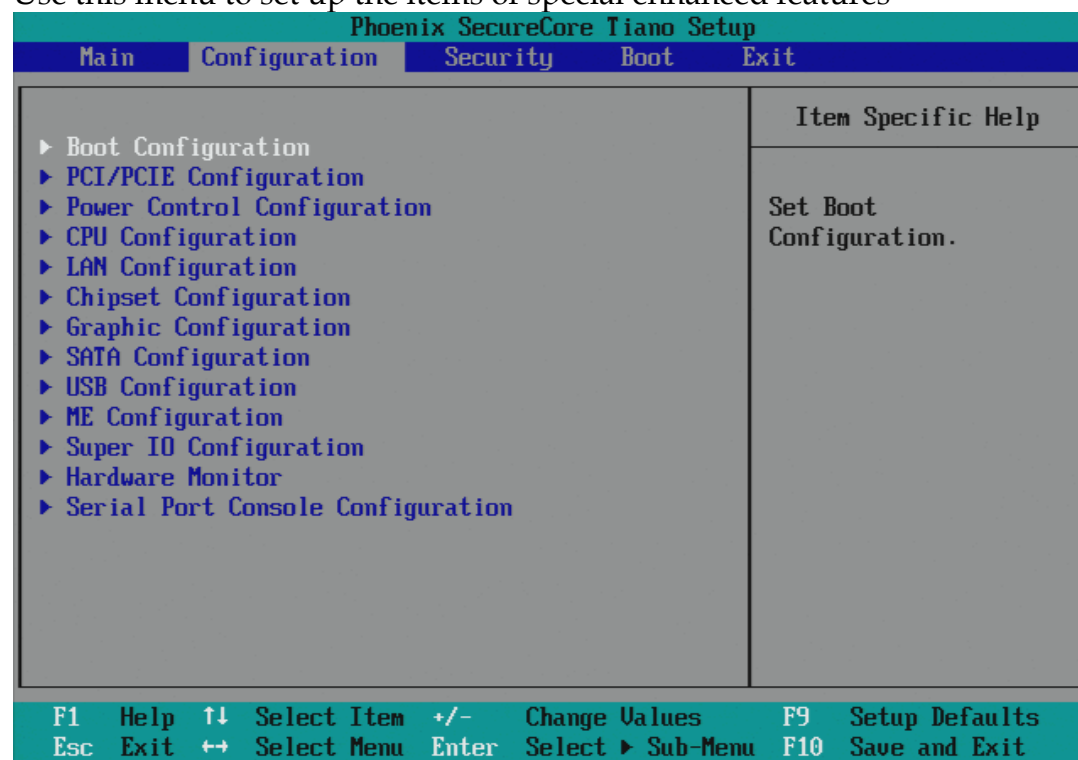
The date format is <Day>, <Month> <Date> <Year>. Use [+] or [–] to configure system Date.

System Time

The time format is <Hour> <Minute> <Second>. Use [+] or [–] to configure system Time.

4.3 Configuration

Use this menu to set up the items of special enhanced features



Boot Configuration

Set Boot Configuration.

Phoenix SecureCore Tiano Setup			
Configuration			
Boot Configuration		Item Specific Help	
NumLock:	[ON]	Selects Power-on state for NumLock.	
Quick Boot	[Disabled]		
Diagnostic Splash Screen	[Disabled]		
Diagnostic Summary Screen	[Enabled]		
Allow Hotkey in S4 resume	[Disabled]		
UEFI Boot	[Disabled]		
F1	Help	↑↓	Select Item
Esc	Exit	↔	Select Menu
+/-	Change Values		
Enter	Select	▶	Sub-Menu
F9	Setup Defaults		
F10	Save and Exit		

NumLock:

Selects Power-on state for NumLock.

Choices: OFF, ON.

Quick Boot

Enable/Disable quick boot.

Choices: Disabled, Enabled.

Diagnostic Splash Screen

If you select 'Enabled' the diagnostic splash screen always displays during boot. If you select 'Disabled' the diagnostic splash screen does not display unless you press HOTKEY during boot.

Choices: Disabled, Enabled.

Diagnostic Summary Screen

Display the Diagnostic summary screen during boot.

Choices: Disabled, Enabled.

UEFI Boot

Enable the UEFI boot.

Choices: Disabled, Enabled.

PCI/PCIE Configuration

Configure PCI/PCI Express slot.

Phoenix SecureCore Tiano Setup	
Configuration	
PCI/PCIE Configuration	
PCI Device Setting	
SERR# Generation	[Disabled]
PCI Express Setting	
▶ Processor PCI Express Configuration	
▶ PCH PCI Express Configuration	
F1 Help ↑↓ Select Item +/- Change Values F9 Setup Defaults Esc Exit ↔ Select Menu Enter Select ▶ Sub-Menu F10 Save and Exit	

Processor PCI Express Configuration

Phoenix SecureCore Tiano Setup	
Configuration	
Processor PCI Express Configuration	Item Specific Help
PEG0 PEG0 - Gen X [Gen1] PEG1 PEG1 - Gen X [Gen1] PEG2 PEG2 - Gen X [Gen1] PEG3 PEG3 - Gen X [Gen1]	Configure PEG0 B0:D1:F0 Speed.
F1 Help ↑↓ Select Item +/- Change Values F9 Setup Defaults Esc Exit ↔ Select Menu Enter Select ▶ Sub-Menu F10 Save and Exit	

FEG0 PEG0 – Gen X

Configure PFG0 B0:D0:F0 Speed.

Choices: Gen1, Gen2, Gen3

FEG1 PEG1 – Gen X

Configure PFG1 B0:D0:F1 Speed.

Choices: Gen1, Gen2, Gen3

FEG2 PEG2 – Gen X

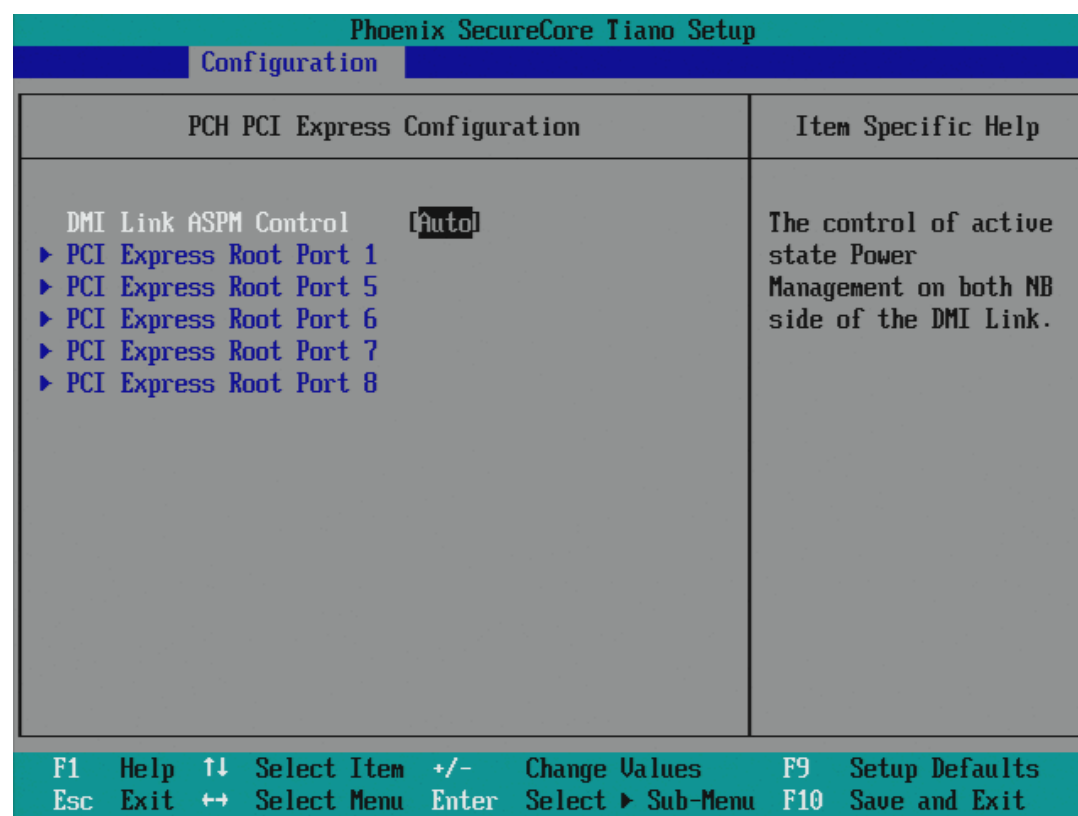
Configure PFG2 B0:D0:F2 Speed.

Choices: Gen1, Gen2, Gen3

FEG3 PEG3 – Gen X

Configure PFG3 B0:D0:F3 Speed.

Choices: Gen1, Gen2, Gen3

ICH PCI Express Configuration**DMI Link ASPM Control**

The control of active state Power Management on both NB side of the DMI Link.

Choices: Disabled, Enabled.

PCI Express Root Port 1, 5-8

Control PCI Express root port.

Phoenix SecureCore Tiano Setup		
Configuration		
PCI Express Root Port 1		Item Specific Help
PCI Express Root Port 1	[Enabled]	Control PCI Express root port.
PCIe Speed	[Auto]	
ASPM	[Disabled]	
HOT PLUG	[Disabled]	
URR	[Disabled]	
FER	[Disabled]	
NFER	[Disabled]	
CER	[Disabled]	
SEFE	[Disabled]	
SENF	[Disabled]	
SECE	[Disabled]	
PME Interrupt	[Disabled]	
PME SCI	[Enabled]	
F1 Help ↑↓ Select Item +/- Change Values F9 Setup Defaults Esc Exit ↔ Select Menu Enter Select ► Sub-Menu F10 Save and Exit		

PCI Express Root Port 1, 5-8

Control PCI Express root port.

Choices: Disabled, Enabled.

PCIe Speed

Select PCIe Speed to Gen1 or Gen2

Choices: Auto, Gen1, Gen2.

ASPM

Control PCIe Active State Power Management settings.

Choices: Disabled, L0S, L1, L0S And L1, Auto.

HOT PLUG

PCI Express Hot Plug Enabled/Disabled.

Choices: Disabled, Enabled.

URR

PCI Express Unsupported Request Reporting Enable/Disable.

Choices: Disabled, Enabled.

FER

PCI Express Device Fatal Error Reporting Enable/Disable.

Choices: Disabled, Enabled.

NFER

PCI Express Device Non-Fatal Error Reporting Enable/Disable.

Choices: Disabled, Enabled.

CER

PCI Express Device Correctable Error Reporting Enable/Disable.

Choices: Disabled, Enabled.

SEFE

Root PCI Express System Error on Fatal Error Enable/Disable.

Choices: Disabled, Enabled.

SENFE

Root PCI Express System Error on Non-Fatal Error Enable/Disable.

Choices: Disabled, Enabled.

SECE

Root PCI Express System Error on Correctable Error Enable/Disable.

Choices: Disabled, Enabled.

PME Interrupt

Root PCI Express PME Interrupt Enable/Disable.

Choices: Disabled, Enabled.

PME SCI

PCI Express PME SCI

Choices: Disabled, Enabled.

Power Control Configuration

Configure ACPI and RTC wake up setting.

Phoenix SecureCore Tiano Setup		
Configuration		
Power Control Configuration		Item Specific Help
ACPI Sleep State	[S3]	Enable or disable System wake on alarm event. When enabled, System will wake on the hr::min::sec specified.
Restore AC power loss	[Power Off]	
SLP_S4 Assertion stretch Enable	[Disabled]	
Wake system with Fixed Time	[Enabled]	
Wake up Date(of Month)	[Everyday]	
Wake up hour	[0]	
Wake up minute	[0]	
Wake up second	[0]	
Wake up By PS/2 Keyboard	[Disabled]	
Wake up By PS/2 Mouse	[Disabled]	
Wake up By PCI	[Enabled]	
Wake up By PCIe	[Enabled]	
Wake up By Ring	[Enabled]	
F1 Help ↑↓ Select Item +/- Change Values F9 Setup Defaults Esc Exit ↔ Select Menu Enter Select ► Sub-Menu F10 Save and Exit		

ACPI Sleep State

Select the highest ACPI Sleep state the system will enter when the SUSPEND button is pressed.

Choices: S1, S3.

Restore AC Power Loss

Select AC Power state when power is re-applied after a power failure.

Choices: Power Off, Power On, Last State.

Wake system with Fixed Time

Enable or disable system wake on alarm event. When Enabled, system will wake on the hr::min::sec specified.

Choices: Disabled, Enabled

Wake up Date (of Month)

If resume by alarm set to enable, select the desired time to wake up system.

Choices: Every day, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31.

Wake up hour

Choices: 0-23

Wake up minute

Choices: 0-59

Wake up second

Choices: 0-59

Wake up By PS/2 Keyboard

Choices: Enabled, Disabled.

Wake up By PS/2 Mouse

Choices: Enabled, Disabled.

Wake up By Ring

Choices: Enabled, Disabled.

CPU configuration

Configure the specific active core(s) and advanced processor management technologies.

Phoenix SecureCore Tiano Setup		
Configuration		
CPU Configuration		Item Specific Help
Max Processor Speed	2.90 GHz	Enabled for windows XP and Linux (OS optimized for Hyper-threading Technology) and Disabled for other OS (OS not optimized for Hyper-Threading). When Disabled only one.
Processor Cores	[2]	
Intel HT technology	Supported	
Hyper-threading	[Enabled]	
Active Processor Cores	[All]	
Limit CPUID Maximum	[Disabled]	
Execute Disable Bit	[Enabled]	
EIST	[Enabled]	
VT-x	[Enabled]	
Local x2APIC	[Disabled]	
F1 Help ↑↓ Select Item +/- Change Values F9 Setup Defaults Esc Exit ↔ Select Menu Enter Select ► Sub-Menu F10 Save and Exit		

Hyper-Threading

Enabled for Windows XP and Linux (OS optimized for Hyper-Threading Technology) and Disabled for other OS (OS not optimized for Hyper-Threading Technology). When Disabled only one.

Choices: Disabled, Enabled.

Active Processor Cores

Select the number of physical cores to enable in each processor package.

Choices: All, 1.

Limit CUID Maximum

Disabled for Windows XP.

Choices: Disabled, Enabled.

Execute Disabled Bit

Enabled Execute Disabled functionality. Also known as Data Execution Prevention (DEP).

Choices: Disabled, Enabled.

EIST

Choices: Disabled, Enabled.

VT-x

When enabled, a VWM can utilize the additional hardware capabilities provided by Vanderpool Technology.

Choices: Disabled, Enabled.

Local x2APIC

Choices: Disabled, Enabled.

LAN Configuration

Configure onboard LAN device.

Phoenix SecureCore Tiano Setup							
Configuration							
LAN Configuration		Item Specific Help					
Intel 82579LM GbE LAN	[Enabled]	Enabled/Disabled Intel 82579LM GbE LAN.					
Intel 82583 LAN	[Enabled]						
Wake on LAN	[Enabled]						
LAN Boot ROM	[Disabled]						
F1	Help	↑↓	Select Item	+/-	Change Values	F9	Setup Defaults
Esc	Exit	↔	Select Menu	Enter	Select ► Sub-Menu	F10	Save and Exit

Intel 82579LM GbE LAN

Enabled/Disabled Intel 82579LM GbE LAN.

Choices: Disabled, Enabled.

Intel 82583 LAN

Enabled/Disabled Intel 82583 LAN.

Choices: Disabled, Enabled.

Wake on LAN

Enabled/Disabled Wake on LAN Function.

Choices: Disabled, Enabled.

LAN Boot ROM

Control LAN Boot ROM (PXE) function.

Choices: Disabled, Enabled.

Chipset Configuration

Configure Chipset provide feature.

Phoenix SecureCore Tiano Setup	
Configuration	
Chipset Configuration	Item Specific Help
VT-d [Disabled] ▶ NB PCIe Configuration ▶ Memory Configuration	Check to enable VT-d function on MCH.
F1 Help ↑↓ Select Item +/- Change Values F9 Setup Defaults Esc Exit ↔ Select Menu Enter Select ▶ Sub-Menu F10 Save and Exit	

VT-d

Check to enable VT-d function on MCH.

Choices: Disabled, Enabled.

NB PCIe Configuration

Config NB PCI Express Settings.

Phoenix SecureCore Tiano Setup	
Configuration	
NB PCIe Configuration	Item Specific Help
Always Enable PEG Disabled PEG ASPM [Auto]	To Enable the PEG Slot.
F1 Help ↑↓ Select Item +/- Change Values F9 Setup Defaults Esc Exit ↔ Select Menu Enter Select ► Sub-Menu F10 Save and Exit	

Always Enable PEG

To Enable the PEG Slot.

Choices: Disabled, Enabled.

PEG ASPM

Control ASPM Support for the PEG Device. This has mp effect if PEF is not the current active service.

Choices: Disabled, Auto, ASPM L0s, ASPM L1, ASPM L0sL1.

Memory Configuration

Memory Configuration Parameters.

Phoenix SecureCore Tiano Setup	
Configuration	
Memory Configuration	Item Specific Help
Memory Frequency 1333 MHz DIMM#0 DIMM#1 DIMM#2 DIMM#3 2048 MB (DDR3) Memory Frequency [Auto]	Maximum Memory Frequency Selections in Mhz.
F1 Help ↑↓ Select Item +/- Change Values Esc Exit ↔ Select Menu Enter Select ► Sub-Menu	F9 Setup Defaults F10 Save and Exit

Memory Frequency

Maximum Memory Frequency Selections in Mhz.

Choices: Auto, 1067, 1333, 1600, 1867, 2133.

Graphic Configuration

Configure integrated Graphic like Boot display, video memory and external Graphic feature.

Phoenix SecureCore Tiano Setup		
Configuration		
Graphic Configuration		Item Specific Help
Primary Display	[Auto]	Select which of IGFX/PEG/PCI Graphics device should be Primary Display Or select SG for Switchable Gfx.
Internal Graphics	[Auto]	
Aperture Size	[256MB]	
DVMT Pre-Allocated	[64M]	
DVMT Total Gfx Mem	[256MB]	
Primay Boot display	[VBIOS Default]	
F1 Help ↑↓ Select Item +/- Change Values F9 Setup Defaults Esc Exit ↔ Select Menu Enter Select ► Sub-Menu F10 Save and Exit		

Primary Display

Select which of IGFX/PEG/PCI Graphics device should be Primary Display Or select SG for Switchable Gfx.

Choices: Auto, IGFX, PEG, PCI.

Internal Graphics

Keep IGD Enabled Based on the setup options,

Choices: Auto, Disabled, Enabled.

Aperture Size

Select the Aperture Size

Choices: 128MB, 256MB, 512MB.

DVMT Pre-Allocated

Select DVMT 5.0 Pre-Allocated Graphics Memory size used by the Internal Graphics Device.

Choices: 0M, 32M, 64M, 96M, 128M, 160M, 192M, 224M, 256M, 288M, 320M , 352M, 384M, 416M, 448M, 480M, 512M.

DVMT Total Gfx Mem

Select DVMT 5.0 Total Graphics Memory size used by the Internal Graphics Device.
Choices: 0-9.

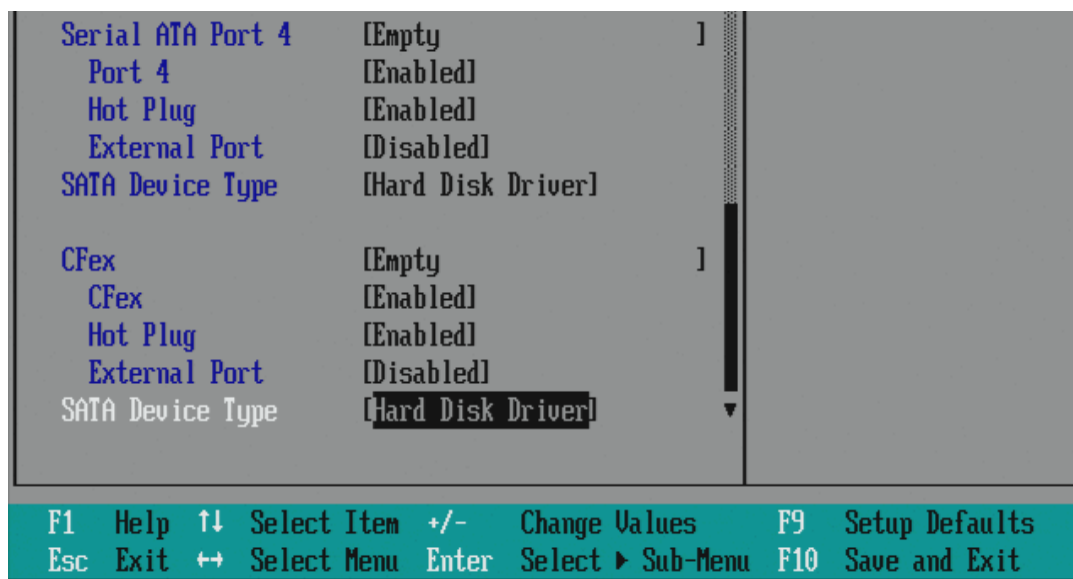
Primay Boot display

Choices: VBIOS Default, VGA, DVI, HDMI.

SATA Configuration

Configure SATA controller and view detected HDD Information.

Phoenix SecureCore Tiano Setup			
Configuration			
SATA Configuration			Item Specific Help
SATA Controller(s)	[Enabled]	▲	Determines how SATA controller(s) operate.
Launch Storage OpROM	[Disabled]		
SATA Mode	[AHCI]		
Serial ATA Port 0	[Empty]	]	
Port 0	[Enabled]		
Hot Plug	[Enabled]		
External Port	[Disabled]		
SATA Device Type	[Hard Disk Driver]		
Serial ATA Port 1	[Empty]	]	
Port 1	[Enabled]		
Hot Plug	[Enabled]		
External Port	[Disabled]		
SATA Device Type	[Hard Disk Driver]	▼	
Serial ATA Port 2	[Empty]	]	
Port 2	[Enabled]		
Hot Plug	[Enabled]		
External Port	[Disabled]		
SATA Device Type	[Hard Disk Driver]		Display the identity of the device attached.
Serial ATA Port 3	[Empty]	]	
Port 3	[Enabled]		
Hot Plug	[Enabled]		
External Port	[Disabled]		
SATA Device Type	[Hard Disk Driver]		



SATA Controller (s)

Determines how SATA controllers (s) operate.

Choices: Disabled, Enabled.

Launch Storage OpROM

Enable or Disable Boot Option for Legacy Mass Storage Devices with Option ROM.

Choices: Enabled, Disabled.

SATA Mode

Determines how SATA controllers (s) operate.

Choices: Disabled, IDE, AHCI, RAID.

Serial ATA Port 0-4, CFex

Display the identity of the device attached.

Choices: Empty.

Port 0-4, CFex

Enabled or Disabled SATA Port.

Choices: Disabled, Enabled.

Hot Plug

Designates this port as Hot Pluggable.

Choices: Disabled, Enabled.

External Port

External SATA Support.

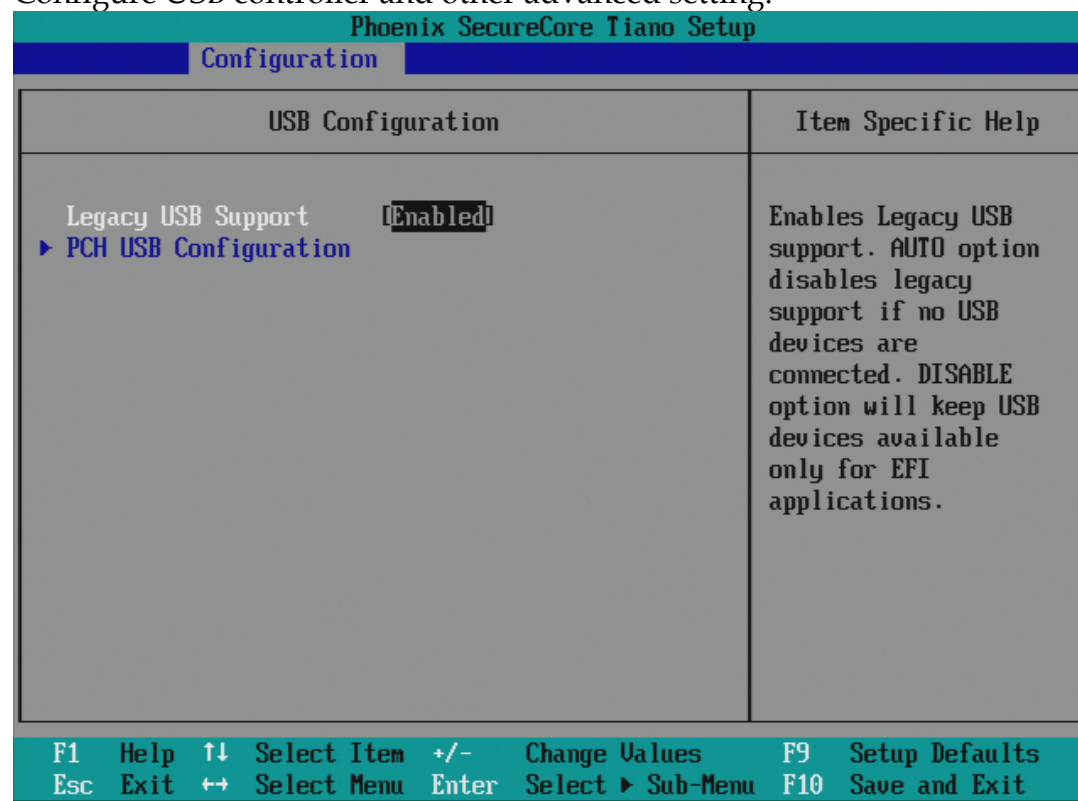
Choices: Disabled, Enabled.

SATA Device Type

Identify the SATA port is connected to Solid State Drive or Hard Disk Drive.
Choices: Hard Disk Drive, Solid State Driver.

USB Configuration

Configure USB controller and other advanced setting.



Legacy USB Support

Enables Legacy USB support. AUTO option disables legacy support if no USB devices are connected. DISABLE option will keep USB devices available only for EFI applications.

Choices: Enabled, Disabled.

PCH USB Configuration

Control each of the USB ports disabling.

Choices: Enabled, Disabled.

Phoenix SecureCore Tiano Setup	
Configuration	
PCH USB Configuration	Item Specific Help
USB Ports Per-Port Disable [Disabled]	Control each of the USB ports disabling.
F1 Help ↑↓ Select Item +/- Change Values F9 Setup Defaults Esc Exit ↔ Select Menu Enter Select ► Sub-Menu F10 Save and Exit	

ME Configuration

Configure Management Engine Technology Parameters.

Phoenix SecureCore Tiano Setup	
Configuration	
ME Configuration	Item Specific Help
ME FW Version 8.1.0.1265 ME Firmware Intel(R) ME 5MB firmware Intel(R) ME [Enabled]	Enable/Disable Intel(R) Management Engine.
F1 Help ↑↓ Select Item +/- Change Values F9 Setup Defaults Esc Exit ↔ Select Menu Enter Select ► Sub-Menu F10 Save and Exit	

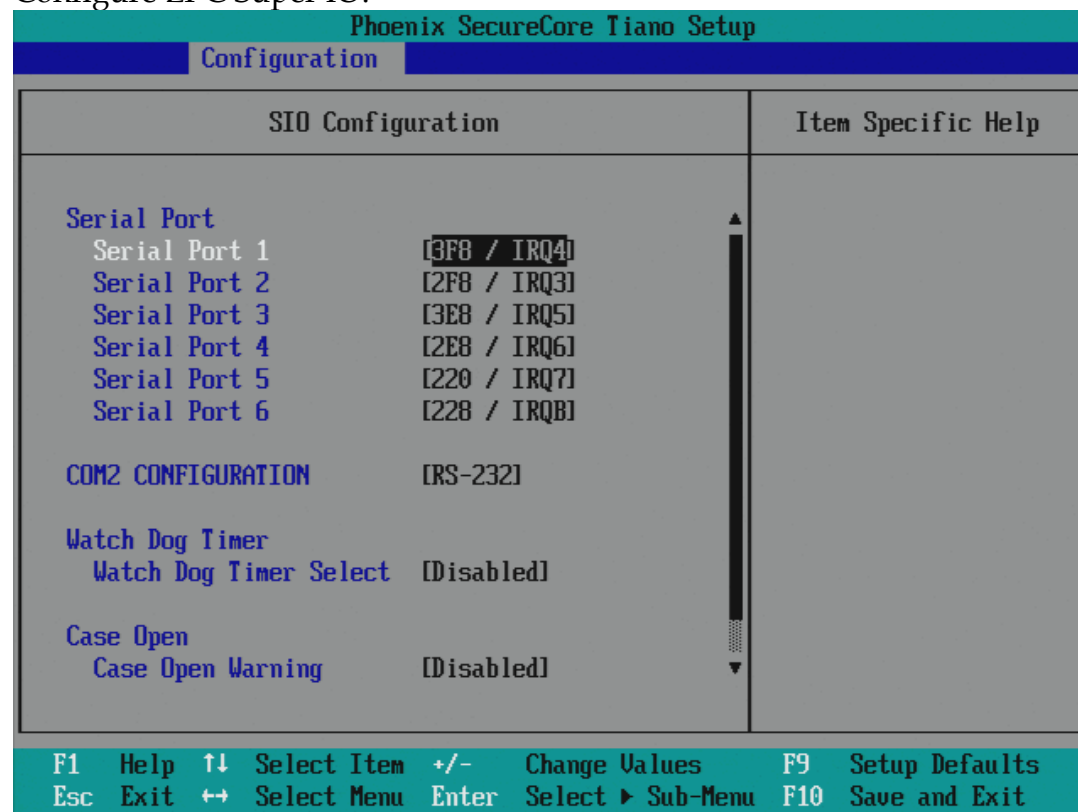
Intel (R) ME

Enable/Disable Intel (R) Management Engine.

Choices: Enabled, Disabled.

Super IO Configuration

Configure LPC Super IO.

**Serial Port 1-6**

Choices: Disabled, 3F8/IRQ4, 2F8/IRQ3, 3F8/IRQ5, 2F8/IRQ6, 220/IRQ7, 228/IRQB.

COM2 Configuration

Select Com2 Configuration.

Choices: RS-232, RS422, RS485.

Watch Dog Timer Select

Choices: Disabled, 15 secs, 30 secs, 1 min, 2 mins, 3 mins.

Case Open

Enable or Disable Case Open Warning Message.

Choices: Enabled, Disabled.

Hardware Monitor

Provide on board sensor reading information. (Show only)

Phoenix SecureCore Tiano Setup	
Configuration	
Hardware Monitor	
▶ CPU Fan Feature	
CPU Temperature	[41 °C]
System Temperature	[33 °C]
CPU Fan	[10384 RPM]
System Fan	[N/A]
Vcore	[0.94 V]
1.5V	[1.47 V]
VCC3	[3.23 V]
VCC	[4.97 V]
+12V	[12.09 V]
-12V	[-11.90 V]
5VSB	[4.99 V]

F1	Help	↑↓	Select Item	+/-	Change Values	F9	Setup Defaults
Esc	Exit	↔	Select Menu	Enter	Select ▶ Sub-Menu	F10	Save and Exit

CPU Fan Feature

Phoenix SecureCore Tiano Setup	
Configuration	
Smart Fan Control Feature	
CPU Fan Control	
Smart Fan Control	[Enabled]
Smart Fan Start Temp	[45 °C]
Fan Full Speed Temp	[85 °C]

F1	Help	↑↓	Select Item	+/-	Change Values	F9	Setup Defaults
Esc	Exit	↔	Select Menu	Enter	Select ► Sub-Menu	F10	Save and Exit

Smart Fan Control

Choices: Enabled, Disabled.

Smart Fan Start Temp

Choices: 25 °C, 30 °C, 35 °C, 40 °C, 45 °C.

Fan Full Speed Temp

Choices: 50 °C, 55 °C, 60 °C, 65 °C, 70 °C, 75 °C, 80 °C, 85 °C.

.

Serial Port Console Configuration

Configure console redirection on serial port.

Phoenix SecureCore Tiano Setup	
Configuration	
Serial Port Console Configuration	Item Specific Help
Serial Port 1 Console Redirection [Disabled] ► Console Redirection Setting	Control Console Redirection enable/disable.
F1 Help ↑↓ Select Item +/- Change Values F9 Setup Defaults Esc Exit ↔ Select Menu Enter Select ► Sub-Menu F10 Save and Exit	

Console Redirection

Control Console Redirection enable/disable.

Choices: Enabled, Disabled

Console Redirection Setting

Phoenix SecureCore Tiano Setup	
Configuration	
Console Redirection Setting	Item Specific Help
Terminal Type [UT100] Bits per second [115200] Flow Control [None]	Control Console Redirection enable/disable.
F1 Help ↑↓ Select Item +/- Change Values F9 Setup Defaults Esc Exit ↔ Select Menu Enter Select ► Sub-Menu F10 Save and Exit	

Terminal Type

Control Console Redirection enable/disable.

Choices: VT100, VT100+, VT-UTF8, ANSI.

Bits per second

Selects serial port transmission speed. The speed must be matched on the other side. Long or noisy lines may require lower speeds.

Choices: 9600, 19200, 57600, 115200.

Flow Control

Set flow control method for UCR.

None- No flow control.

RTS/CTS - Hardware flow control.

XON/XOFF - Software flow control

Choices: Hardware RTS/CTS, Software Xon/Xoff.

4.4 Security

This section lets you set security passwords to control access to the system at boot time and/or when entering the BIOS setup program.

Phoenix SecureCore Tiano Setup		
Main	Configuration	Security
Supervisor Password is: Cleared User Password is: Cleared Set Supervisor Password [Enter] Supervisor Hint String [] Set User Password [Enter] User Hint String [] Min. password length [1] Authenticate User on Boot [Disabled] HDD Security Status No HDD detected Trusted Platform Module (TPM)		Item Specific Help Set or clear the Supervisor account's password.

F1	Help	↑↓	Select Item	+/-	Change Values	F9	Setup Defaults
Esc	Exit	↔	Select Menu	Enter	Select ▶ Sub-Menu	F10	Save and Exit

Set Supervisor Password

Set or clear the Supervisor account's password.

Supervisor Hint String

Press Enter to type Supervisor Hint String.

Set User Password

Set or clear the User account' password.

Supervisor Hint String

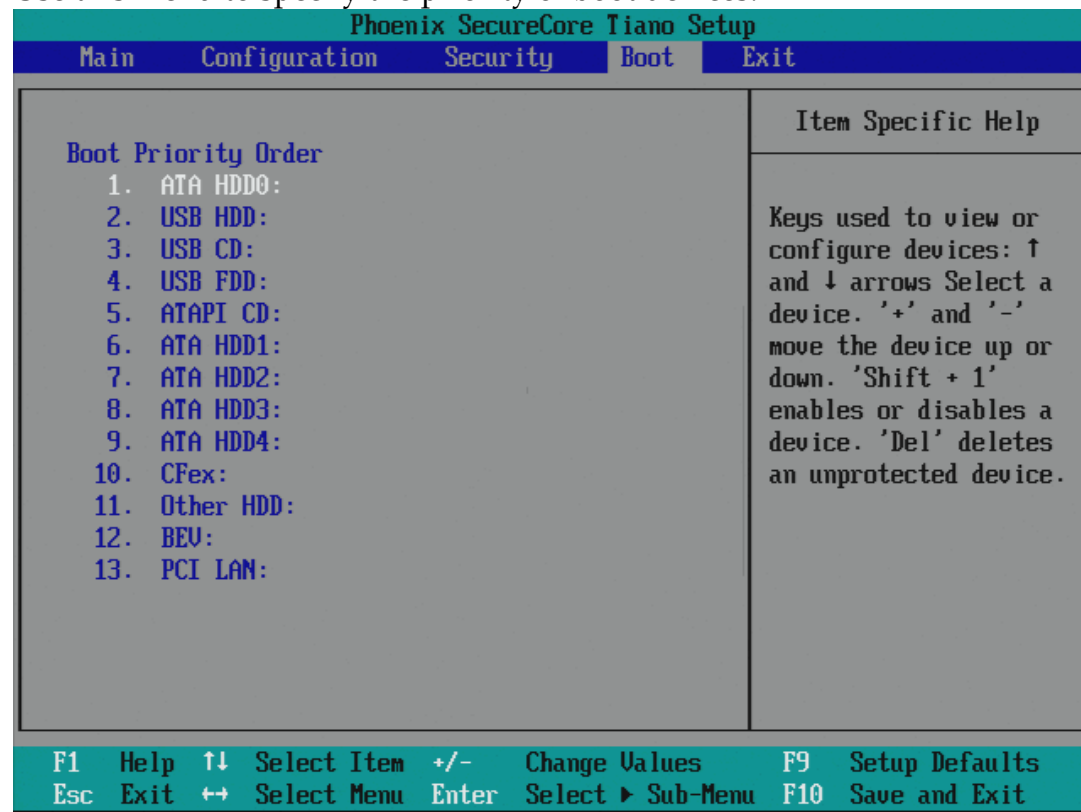
Press Enter to type User Hint String.

Min. password length

Set the minimum number of characters for password (1-20).

4.5 Boot

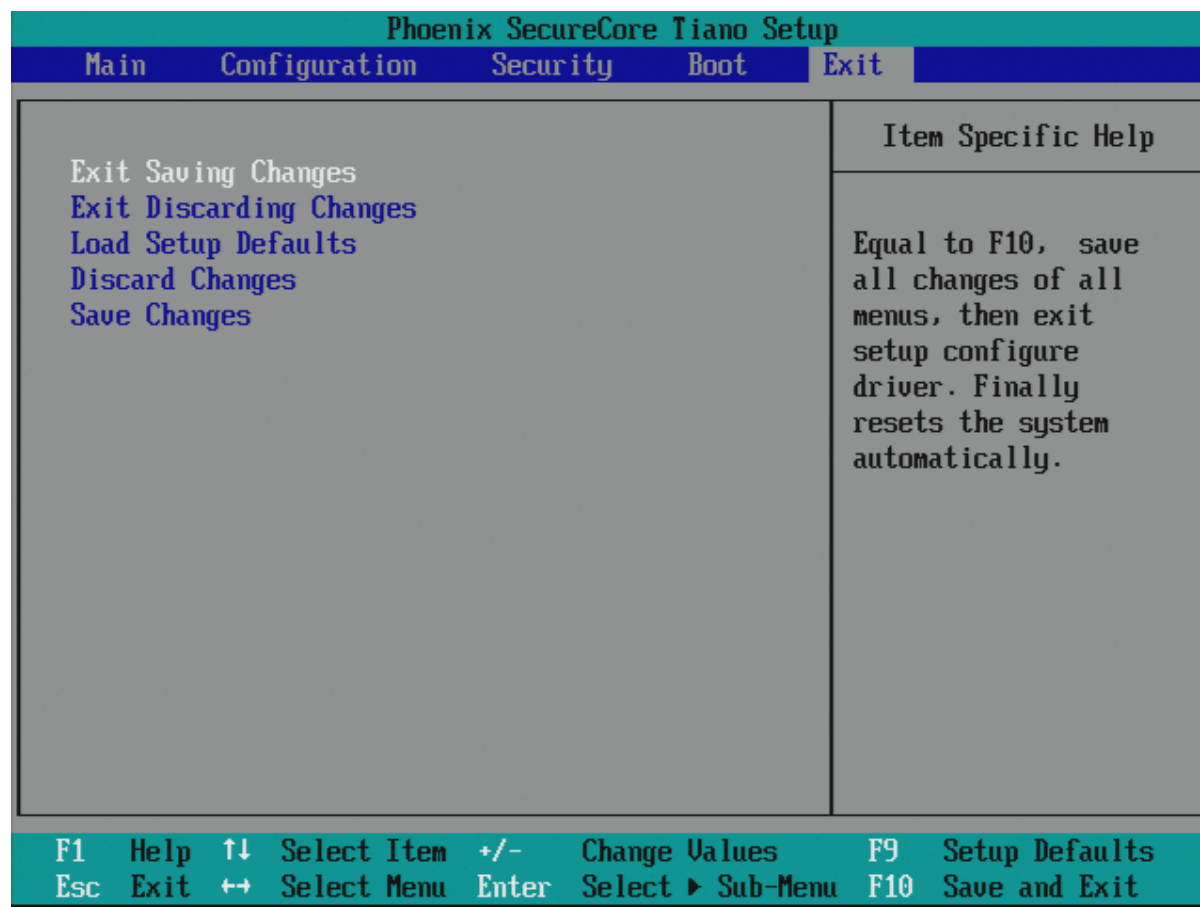
Use this menu to specify the priority of boot devices.



Boot Priority Order

Keys used to view or configure devices: ↑ and ↓ arrows Select a device. '+' and '-' move the device up or down. 'Shift + 1' enabled or disables a device. 'Del' deletes an unprotected device.

4.6 Exit



Exit Saving Changes

Equal to F10, save all changes of all menus, then exit setup configure driver. Finally resets the system automatically.

Exit Discarding Changes

Equal to ESC, never save changes, then exit setup configure driver.

Load Setup Defaults

Equal to F9. Load standard default values.

Load Optimized Defaults

Load settings for optimized boot time and system performance.

Discard Changes

Load the original value of this boot time. Not the default Setup value.

Save Changes

Save all changes of all menus, but do not reset system.

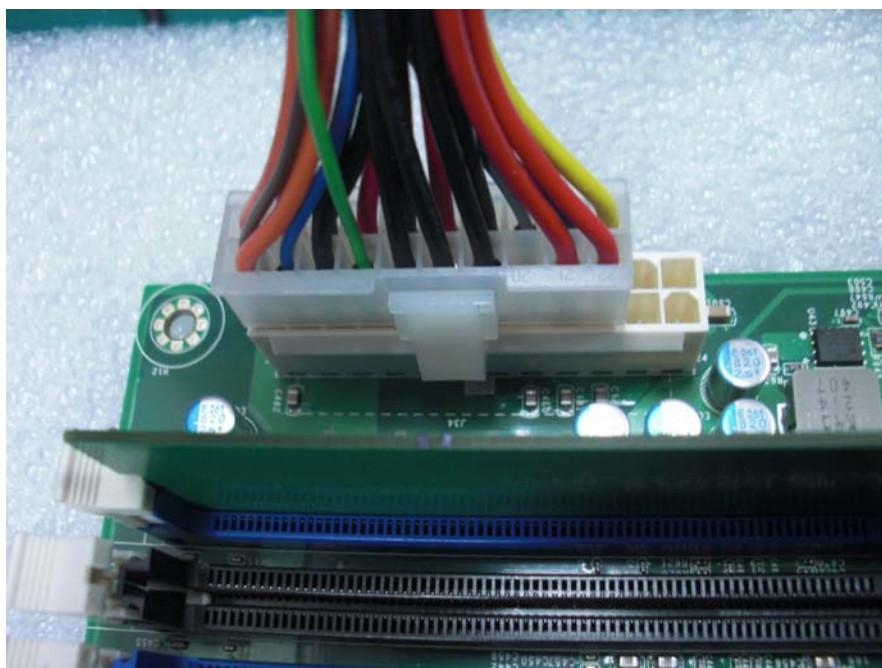
Chapter 5

Troubleshooting

This chapter provides a few useful tips to quickly get RUBY-D714 running with success. As basic hardware installation has been addressed in Chapter 2, this chapter will focus on system integration issues, in terms of BIOS setting, and OS diagnostics.

5.1 Hardware Quick Installation

ATX Power Setting



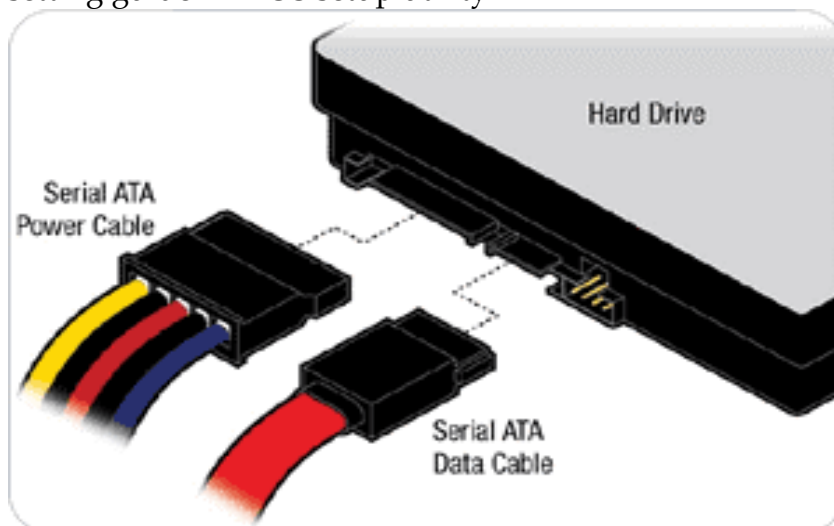
RUBY-D714 supports ATX. Therefore, there is no other setting that really needs to be set up. However, there are only two connectors that must be connected – J8 (4 pins CPU +12V main power connector) & J7 (16 pins ATX Power Connector)

ATX Power Connector 4-Pins CPU Main Power Connector



Serial ATA Hard Disk Setting for IDE/RAID/AHCI

Unlike IDE bus, each Serial ATA channel can only connect to one SATA hard disk at a time; there are total six connectors, SATA1~6 port. The installation of Serial ATA is simpler and easier than IDE, because SATA hard disk doesn't require setting up Master and Slave, which can reduce mistake of hardware installation. All you need to operate IDE, RAID (0/1/5/10) and AHCI application for system, please follow up setting guide in BIOS setup utility



5.2 BIOS Setting

It is assumed that users have correctly adopted modules and connected all the devices cables required before turning on ATX power. CPU, CPU Fan, 204-pin DDR3 memory, keyboard, mouse, floppy drive, SATA hard disk, DVI-I connector, but it only can use on DVI-D function, doesn't support DVI-I function, device power cables,

ATX accessories are good examples that deserve attention. With no assurance of properly and correctly accommodating these modules and devices, it is very possible to encounter system failures that result in malfunction of any device.

To make sure that you have a successful start with RUBY-D714, it is recommended, when going with the boot-up sequence, to hit "DEL" key and enter the BIOS setup menu to tune up a stable BIOS configuration so that you can wake up your system far well.

Loading the default optimal setting

When prompted with the main setup menu, please scroll down to "Load Optimal Defaults", press "Enter" and "Y" to load in default optimal BIOS setup. This will force your BIOS setting back to the initial factory configuration. It is recommended to

do this so you can be sure the system is running with the BIOS setting that Portwell has highly endorsed. As a matter of fact, users can load the default BIOS setting any time when system appears to be unstable in boot up sequence.

Auto Detect Hard Disks

In the BIOS => Standard CMOS setup menu, pick up any one from Primary/Secondary Master/Slave IDE ports, and press "Enter". Setup the selected IDE port and its access mode to "Auto". This will force system to automatically pick up the IDE devices that are being connected each time system boots up.

Improper disable operation

There are too many occasions where users disable a certain device/feature in one application through BIOS setting. These variables may not be set back to the original values when needed. These devices/features will certainly fail to be detected.

When the above conditions happen, it is strongly recommended to check the BIOS settings. Make sure certain items are set as they should be. These include the COM1/

COM2 ports, USB ports, external cache, on-board VGA and Ethernet.

It is also very common that users would like to disable a certain device/port to release IRQ resource. A few good examples are

Disable COM1 serial port to release IRQ #4

Disable COM2 serial port to release IRQ #3

Etc...

A quick review of the basic IRQ mapping is given below for your reference.

Interrupt Request Lines IRQ		
IRQ#	Current Use	Default Use
IRQ 0	Unused	System Timer
IRQ 1	System ROM	Keyboard Event
IRQ 2	【Unassigned】	Usable IRQ
IRQ 3	【Unassigned】	Usable IRQ
IRQ 4	【Unassigned】	Usable IRQ
IRQ 5	【Unassigned】	Usable IRQ
IRQ 6	System ROM	Diskette Event
IRQ 7	Unused	Usable IRQ
IRQ 8	System ROM	Real-Time Clock
IRQ 9	【Unassigned】	Usable IRQ
IRQ 10	【Unassigned】	Usable IRQ
IRQ 11	【Unassigned】	Usable IRQ
IRQ 12	System ROM	IBM Mouse Event
IRQ 13	System ROM	Coprocessor Error
IRQ 14	System ROM	Hard Disk Event
IRQ 15	【Unassigned】	Usable IRQ

It is then very easy to find out which IRQ resource is ready for additional peripherals. If IRQ resource is not enough, please disable some devices listed above to release further IRQ numbers.

5.3 FAQ

Installation Problem

Question: How to update the BIOS file of the RUBY-D714?

Answer:

Please visit web site of the Portwell download center as below hyperlink and register an account. <http://www.portwell.com.tw/support/> Input your User name and password to log in the download center. Select the "Search download" to input the keyword "RUBY-D714". Find the "BIOS" page to download the ROM file and flash utility.

Execute the zip file to root of the bootable USB Pen drive.

Insert your bootable USB Pen drive in WADE-8020 board and power-on.

Input the "FPT /f XXXXX.ROM /BIOS" to start to update BIOS. ("XXXXX" is the file name of the ROM file.)

Switch "Off" the Power Supply when you finished the update process.

To short the JP1 2-3 jumper for 5 seconds then set back to normal. (Clear CMOS)

Switch "ON" the Power Supply then press the "del" key to BIOS to load "Restore Defaults" then save them to exit.

Note:

Please visit our technical web site at <http://www.portwell.com.tw>

For additional technical information, which is not covered in this manual, you can mail to tsd@mail.portwell.com.tw or to our sales for further assistance. Thank you.

5.4 System Memory Address Map

Each On-board device in the system is assigned a set of memory addresses, which also can be identical of the device. The following table lists the system memory address used for your reference.

System Memory Address Map		
Memory Area	Size	Description
0000-003F	1K	Interrupt Area
0040-004F	0.3K	BIOS Data Area
0050-006F	0.5K	System Data
0070-0E2E	54K	DOS
0E2F-0F6B	5K	Program Area
0F6C-9B7F	560K	【Available】
9B80-9D7F	8K	Unused
First Meg -- Conventional memory end at 630K --		
9D80-9DFF	2K	Extended BIOS Area
9E00-9FFF	8K	Unused
A000-AFFF	64K	VGA Graphics
B000-B7FF	32K	Unused
B800-BFFF	32K	VGA Text
C000-CE3F	57K	Video ROM
CE40-EFFF	135K	Unused
F000-FFFF	64K	System ROM
HMA	64K	First 64K Extended